

Nele Rußwinkel | Uwe Drewitz | Hedderik van Rijn (eds.)

Proceedings of

ICCM 2012

11th International Conference on Cognitive Modeling

ISBN 978-3-7983-2408-4 (Online-Version)

Berlin 2012

Vertrieb/ Universitätsverlag der TU Berlin
Publisher: Universitätsbibliothek
Fasanenstr. 88 (im VOLKSWAGEN-Haus), D-10623 Berlin
Tel.: (030) 314-76131; Fax: (030) 314-76133
E-Mail: publikationen@ub.tu-berlin.de
<http://www.univerlag.tu-berlin.de>

Table of Contents

Organization & Committees	vii
Sponsoring & Support	ix
Invited Speakers	x
- Niels Taatgen	xi
- Klaus Oberauer	xii
- Alex Kirlik	xiii
Tutorials	xiv
Papers & Abstracts	xxii
Talks: Decision Making, Prototypes, and Analogies	
<i>Modeling Intuitive Decision Making in ACT-R</i>	1
William Kennedy, Robert Patterson	
<i>Dynamic Use of Multiple Analogies in the AMBR Model Causing Re-Representation of the Target</i>	7
Georgi Petkov, Ivan Vankov, Boicho Kokinov	
<i>A Cognitive Model For Predicting Aesthetical Judgements As Similarity to Dynamic Prototypes</i> ..	13
Ute Schmid, Michael Siebers, Johannes Folger, Simone Schineller, Dominik Seuss, Marius Raab, Claus-Christian Carbon, Stella J. Faerber	
<i>An ACT-R Approach to Reasoning about Spatial Relations with Preferred and Alternative Mental Models</i>	19
Sven Brüßow, Matthias Frorath, Marco Ragni, Thomas Fangmeier	
<i>A multinomial model of applying recognition to judge between multiple alternatives</i>	25
Philip Beaman	
Talks: Learning and Problem Solving	
<i>Data Acquisition Dynamics and Hypothesis Generation</i>	31
Nicholas D. Lange, Rick P. Thomas, Eddy J. Davelaar	
<i>A cognitive architecture-based model of graph comprehension</i>	37
David Peebles	
<i>Cognitive modelling of early music reading skill acquisition for piano</i>	43
Bruno Emond, Gilles Comeau	
<i>An account of cognitive flexibility and inflexibility for a complex dynamic task</i>	49
Alberto De Obeso Orendain, Sharon Wood	

Symposium: Cognitive Modeling of Processes "Beyond Rational"

<i>Cognitive Modeling of Processes "Beyond Rational"</i>	55
William Kennedy, Frank Ritter, Christian Lebiere, Ion Juvina, Alessandro Oltramari, Jonathan Gratch, Richard Young	

Poster Session 1

<i>Computational Simulation Of Visual Distraction Effects On Car Drivers' Situation Awareness ...</i>	59
Thierry Bellet, Jean-Charles Bornard, Pierre Mayenobe, Jean-Christophe Paris, Dominique Gruyer, Bernard Claverie	
<i>SimPilot: An exploration of modeling a highly interactive task with delayed feedback in a multitasking environment</i>	66
Michael Schoelles, Wayne Gray	
<i>Cognitive Re-Use via Emergic Networks</i>	72
David Pierre Leibovitz, Robert West	
<i>Hierarchical Bayesian Modeling of Manipulation Sequences from Bimodal Input</i>	74
Alexandra Barchunova, Jan Moringen, Robert Haschke, Helge Ritter	
<i>Modeling the User's Belief about the State of a Spoken Dialog System</i>	80
Klaus-Peter Engelbrecht	
<i>A four factor model of landmark salience – A new approach</i>	82
Florian Röser, Antje Krumnack, Kai Hamburger, Markus Knauff	
<i>Modeling the Effects of Base-rates on Cyber Threat Detection Performance</i>	88
Varun Dutt, Young-Suk Ahn, Noam Ben-Asher, Cleotilde Gonzalez	
<i>Exploring Feature Collocation for Semantic Concept Identification</i>	94
Meng Yu	
<i>Uses, Abuses and Misuses of Computational Models in Classical Conditioning</i>	96
Alonso Eduardo, Mondragon Plaza	
<i>How WM load influences pronoun interpretation</i>	101
Jacolien van Rij, Hedderik van Rijn, Petra Hendriks	
<i>An ACT-R Model of Credibility Judgment of Micro-blogging Web Pages</i>	103
Q. Vera Liao, Peter Piroli, Wai-Tat Fu	
<i>Computationally Efficient Forgetting via Base-Level Activation</i>	109
Nate Derbinsky, John Laird	
<i>A Change for the Better? Assessing the Computational Cost of Re-Representation</i>	111
Todd Wareham, Robert Robere, Iris van Rooij	
<i>Defining Factors of Interest for Large-Scale Socio-Cognitive Simulations</i>	117
Ryan Kaulakis, Changkun Zhao, Jonathan Morgan, Jeremiah Hiam, Joseph Sanford, Frank Ritter	
<i>Emotion Detection by Event Evaluation using Fuzzy Sets as Appraisal Variables</i>	123
Michael Kotzyba, Barbara Deml, Hendrik Neumann, Stefan Gluege, Kim Hartmann, Ingo Siegert, Andreas Wendemuth, Harald Traue, Steffen Walter	

<i>Modeling behavior and performance of air traffic controllers using coloured Petri nets</i>	125
Hardy Smieszek	
<i>A Framework for Task Accomplishment Using an ACT-R Simulation</i>	127
Michael Beckmann, Ulas Yilmaz, Gloria Pöhler, Anne Wegerich	
<i>Performance Modelling of Interface Dynamism in Motor Skills Acquisition</i>	129
Olurotimi Richard Akinlofa, Patrik O'Brian Holt, Eyad Elyan	
<i>Explorations in ACT-R Based Language Analysis – Memory Chunk Activation, Retrieval and Verification without Inhibition</i>	131
Jerry Ball	
<i>Prospective Memory and Working Memory: A Hierarchical Modeling Approach</i>	137
Nina R. Arnold, Ute J. Bayen, Rebekah E. Smith	
<i>Non Inversion of Similarity and Dissimilarity in Judging Faces</i>	139
Ludovica Lorusso, Enrico Grosso	
<i>Towards optimal payoff manipulations</i>	140
Christian Janssen, Andrew Howes, Duncan Brumby	
<i>What Can (and Can't) Make Problem Solving by Insight Possible? A Complexity-Theoretic Investigation</i>	142
Todd Wareham	
<i>A Concept of a Cognitive System of Motivation</i>	144
Andreas Ammann	
<i>A Computational Model for Forms of Selective Attention based on Cognitive and Affective Feelings</i>	145
Luis Macedo	
<i>A process model of immediate inferences</i>	151
Sangeet Khemlani, Greg Trafton, Max Lotstein, Phil Johnson-Laird	
<i>Cognitive Robotics: Analysis of Preconditions and Implementation of a Cognitive Robotic System for Navigation Tasks</i>	157
Stefano Bennati, Marco Ragni	
<i>Accumulation of Evidence and Information Search in Experiential Decisions</i>	163
Katja Mehlhorn, Varun Dutt, Noam Ben-Asher, Cleotilde Gonzalez	
<i>A Computational Model of Collective Sensemaking: Differential Effects of Communication Network Structure on Collective Sensemaking Abilities</i>	165
Paul Smart	
<i>Using Support Vector Machines to Determine Linear Separability</i>	167
Bram Van Rensbergen, Eef Ameel, Fons Van Assche, Gert Storms	

Talks: Memory and Learning

<i>A Rational Analysis of Memory Search Termination</i>	169
Eddy Davelaar, Erica Yu, J. Isaiah Harbison, Erika Hussey, Michael Dougherty	
<i>Modeling representational shifts in learning the number line</i>	175
Leif Kåre Lende, Niels Taatgen	

<i>Oscillatory Basis of Individual Differences in Working Memory Capacity</i>	181
Krzysztof Andreczyk, Tomasz Smolen, Adam Chuderski	
<i>Modeling the Temporal Dynamics of Visual Working Memory</i>	187
Johannes Lohmann, Oliver Herbolt, Martin V. Butz	
Talks: Architectures, I	
<i>A Reinforcement Learning Model of Bounded Optimal Strategy Learning</i>	193
Xiuli Chen, Andrew Howes	
<i>Bridging the Gap between Theory and Practice of Approximate Bayesian Inference</i>	199
Johan Kwisthout, Iris van Rooij	
<i>Competence-Preserving Retention of Learned Knowledge in Soar’s Working and Procedural Memories</i>	205
Nate Derbinsky, John Laird	
<i>Pre-attentive and Attentive Vision Module</i>	211
Enkhbold Nyamsuren, Niels Taatgen	
Talks: Architectures, II	
<i>ACTR-QN: Integrating Queueing Network and ACT-R Cognitive Architectures</i>	217
Shi Cao, Yili Liu	
Talks: Decomposing Cognition	
<i>Relating ACT-R buffer activation to EEG activity during an attentional blink task</i>	218
Marieke van Vugt	
<i>One Model, Two Tasks: Decomposing the Change Signal Task</i>	224
Larry Moore, Glenn Gunzelmann, Marissa Daigle	
Poster Session 2	
<i>Verbalization of problem solving processes in unaided object assembly</i>	230
Linn Gralla, Thora Tenbrink	
<i>Modeling Risky Decision Making by Cellular Automata</i>	232
Kenryo Indo	
<i>Conditioning for Least Action</i>	234
Eduardo Alonso, Michael Fairbank, Esther Mondragon	
<i>Annual Cognitive Modeling Competition</i>	240
Kevin Gluck	
<i>Choices, choices: task selection preference during concurrent multitasking</i>	241
Menno Nijboer, Niels Taatgen, Hedderik van Rijn	
<i>Extending and Evaluating a Multiplicative Model for Semantic Composition in a Distributional Semantic Model</i>	243
Akira Utsumi	
<i>Towards Pinpointing the Neural Correlates of ACT-R: a Conjunction of Two Model-Based fMRI Analyses</i>	249
Jelmer Borst, John Anderson	

<i>Qualitative and Quantitative Individual Differences in Semantic Categorization</i>	251
Steven Verheyen, Gert Storms	
<i>Modeling Efficiency-guided Modality Choice in Voice and Graphical User Interfaces</i>	253
Stefan Schaffer, David Reitter	
<i>A First Approach to Generate Hybrid Animations for Maintenance Tasks in IPS²</i>	255
Ulrike Schmunzsch, Ulaş Yılmaz, Ralf Reichmuth, Matthias Rötting	
<i>Abstraction in Natural Language Categories</i>	257
Loes Stukken, Wouter Voorspoels, Gert Storms, Wolf Vanpaemel	
<i>Semantic Cognition: A Re-examination of the Recurrent Network "Hub" Model</i>	259
Olivia Guest, Rick Cooper	
<i>Memory and Contextual Change in Causal Learning</i>	265
Uwe Drewitz, Stefan Brandenburg	
<i>Functional Modeling of Personality Properties Based on Motivational Traits</i>	271
Joscha Bach	
<i>Model-Based Prediction of Between-Trial Fluctuations in Response Caution From EEG Data</i> ...	273
Udo Boehm, Leendert van Maanen, Birte Forstmann, Hedderik van Rijn	
<i>Cognitive Model of Drivers Attention</i>	275
Kerstin Haring, Marco Ragni, Lars Konieczny	
<i>Cognitive modeling of different processing modes in task switching: toward an explanation of the effect of aging on switching cost</i>	281
Stephane Deline, Jacques Juhel	
<i>A LIDA-based Model of the Attentional Blink</i>	283
Tamas Madl, Stan Franklin	
<i>Self-Organization and Associative Learning as a Basis for Cognitive and Sensorimotor Modeling of Speech Production, Perception, and Acquisition</i>	289
Bernd Kroeger, Cornelia Eckers, Christiane Neuschaefer-Rube	
<i>The Role of Memory in MHP/RT: Organization, Function and Operation</i>	291
Muneo Kitajima, Makoto Toyota	
<i>Modeling Behavior of Attention-Deficit-Disorder Patients in a N-Back Task</i>	297
Jonathan Kottlors, Daniel Brand, Marco Ragni	
<i>Combining cognitive modeling and EEG to predict user behavior in a search task</i>	303
Felix Putze, Johannes Meyer, Joscha Borné, Tanja Schultz, Daniel Holt, Joachim Funke	
<i>Towards a 50 msec Cognitive Cycle in a Graphical Architecture</i>	305
Paul Rosenbloom	
<i>Practical Optimal-Solution Algorithms for Schema-based Analogy Mapping</i>	311
Robert Hamilton, Todd Wareham	
<i>Extending the Computational Belief-Desire Theory of Emotions to Fantasy Emotions</i>	313
Rainer Reisenzein	

<i>Dual-route Connectionist model of Greek Spelling</i>	315
Ioanna Katidioti, Ian Simpson, Athanassios Protopapas	
<i>Matching Results of Latent Dirichlet Allocation for Text</i>	317
Andreas Niekler, Patrick Jähnichen	
<i>A self-organized neuronal comparator</i>	323
Guillermo Luduena, Claudius Gros	
Symposium: Human Performance Modeling	
<i>Symposium on Human Performance Modeling</i>	324
Wayne Gray, David Kaber, Guk-Ho Gil, Sang-Hwan Kim, Noam Ben-Asher, Joachim Meyer, Shi Cao, Yili Liu, Cleotilde Gonzalez, Glenn Gunzelmann, Kevin Gluck	
Author Index	328
Reviewers List	331

Organizing Committee

Nele Rußwinkel & Uwe Drewitz , Technische Universität Berlin

Hedderik van Rijn, University of Groningen

Tutorials

Frank Ritter, Pennsylvania State University

Program Committee

Eduardo Alonso, City University London

Erik Altmann, Michigan State University

Thierry Bellet, INRETS (France)

Jelmer Borst, Carnegie Mellon University

Duncan Brumby, University College London

Mike Byrne, Rice University

Rick Cooper, Birkbeck University of London

Gary Cottrell, University of California, San Diego

Fabio Del Missier, University of Trento

Uwe Drewitz, Technische Universität Berlin

Wai-Tat Fu, University of Illinois at Urbana-Champaign

Daniilo Fum, Dipartimento di Psicologia

Kevin Gluck, Air Force Research Laboratory

Fernand Gobet, Brunel University

Cleotilde Gonzalez, Carnegie Mellon University

Glenn Gunzelmann, Air Force Research Laboratory

Tim Halverson, Air Force Research Laboratory

Andrew Howes, University of Manchester

Christian Janssen, University College London

Gary Jones, Nottingham Trent University

Mark Keane, University College Dublin

Markus Knauff, University of Giessen

Boicho Kokinov, New Bulgarian University

Joseph Krems, Technische Universität Chemnitz

Bernd Kroeger, RWTH Aachen University

Johan Kwisthout, Radboud University Nijmegen

Peter Lane, University of Hertfordshire

Christian Lebiere, Carnegie Mellon University

Rick Lewis, University of Michigan

Luís Macedo, University of Coimbra

Michael Matessa, ALION

Ralf Mayrhofer, University of Goettingen

Alain Mille, University of Lyon

Claus Moebus, University of Oldenburg

Shane Mueller, Michigan Technological University

Christopher Myers, Air Force Research Laboratory

Josef Nerb, University of Education Freiburg

Hansjoerg Neth, Max-Planck Institute for Human Development

David Noelle, University of California, Merced

David Peebles, University of Huddersfield
Thad Polk, University of Michigan
Marco Ragni, University of Freiburg
Rainer Reisenzein, University of Greifswald
Frank Ritter, Pennsylvania State University
Nele Rußwinkel, Technische Universität Berlin
Dario Salvucci, Drexel University
Ute Schmid, University Bamberg
Mike Schoelles, Rensselaer Polytechnic Institute
Lael Schooler, Max Planck Institute for Human Development
Barry Silverman, University of Pennsylvania
Robert St. Amant, North Carolina State University
Terrence Stewart, University of Waterloo
Andrea Stocco, Carnegie Mellon University
Niels Taatgen, University of Groningen
Julia Taylor, Purdue University
Manfred Thüring, Technische Universität Berlin
Greg Trafton, Naval Research Laboratory
Leon Urbas, Technische Universität Dresden
Leendert van Maanen, University of Amsterdam
Marieke van Vugt, University of Groningen
Robert West, Carleton University
Sharon Wood, University of Sussex
Richard Young, University College London

Tutorials Committee

Frank E. Ritter, Pennsylvania State University
Thierry Bellet, INRETS (France)
Éric Raufaste, University of Toulouse
Jim Davies, Carleton University
Olivier Georgeon, University of Lyon
Randolph M. Jones, Soar Technology
Maik Friedrich, German Aerospace Center
Nele Rußwinkel, Technische Universität Berlin
Josef Nerb, University of Education Freiburg

Organizing Team

Christian Karl
Stefan Lindner
Gloria Pöhler
Juliane Richter
Maria Schinkmann
Fei Yan

Conference Poster and Cover Design

Marlene Vogel

Webmaster

Steffen Hartwig

Sponsoring & Support

The organizers would like to thank all sponsors for their active support and cooperation in the preparation and realization of the conference.

We acknowledge the financial support from:

DFG - Deutsche Forschungsgemeinschaft

(German Research Foundation, www.dfg.de)

European Office of Aerospace Research & Development

(www.london.af.mil)

Gesellschaft für Kognitionswissenschaft e.V.

(German Cognitive Science society, www.gk-ev.de)

Gesellschaft von Freunden der TU Berlin e.V.

(www.freunde.tu-berlin.de)

Cognitive Science Society

(cognitivesciencesociety.org)

Supported by:

Graduiertenkolleg *Prometei*

(research training group *prometei*, www.prometei.de)

Invited Speakers

The Nature and Transfer of Cognitive Skill

Niels Taatgen (niels@ai.rug.nl)

University of Groningen

Is the whole of human cognitive ability an integrated system of knowledge, strategies and skills, or a collection of individual tasks and goals? Even though most people would gravitate towards the former point of view, the tradition of psychology, cognitive science and cognitive modeling adopts at least the research stance of the latter.

The discussion can be traced back to Thorndike, who rejected the idea of a "formal discipline of the mind", and replaced it with the theory of identical elements. According to this theory, any ability we have is largely independent of other abilities, unless the two share identical elements of knowledge (Stimulus-Response bonds in the time of Thorndike). Singley and Anderson introduced the modern version of this idea: transfer between individual skills is only possible if they share identical production rules.

As a consequence, the current research tradition is to study individual skills and tasks with little regard for interactions between tasks. This is reinforced by many studies that demonstrate a lack of transfer, for example the well-known example in which subjects fail to solve a puzzle about a heart surgeon using a laser to remove a tumor after reading a story about a general who uses his army to concur the capital. Or the fact that even after taking a course in logic, students still fail to solve Wason's selection task.

Most examples of failed transfer, however, play out on a semantic level, in which subjects fail to make the appropriate analogy, even if it is almost forced-fed into them. However, there are several experiments that do show transfer, but the transfer of knowledge seems to play out a more mechanical, syntactic level. In those experiments, subjects can perform or learn particular tasks faster because they have already learn a similar other task. Singley and Anderson's experiment with learning text editors is an example: it is easier to learn a new text editor if you have already mastered a different editor.

A new branch of more recent experiments have a similar structure, but focus on executive control. By training a particular control task, for example task switching, N-Back or working memory, subjects also improve on other executive control tasks, like the Stroop task.

Cognitive models have a hard time explaining transfer between skills. In production system models rules are typically specific to a task, and neural networks models are typically also geared towards a particular task. In my talk, I will propose a solution based on the ACT-R architecture that involves breaking down productions into their smallest components, in which a rule is reduced to either a single comparison or a single atomic action. Models constructed on this basis combine these basic components into compound rules that are still independent of the particular task, but that can be used in other tasks that share the same thinking structures. For example, counting can be helpful in learning to reason in a semantic network, because both tasks involve iteration.

I will demonstrate the generality of the approach with a set of examples (and as time permits): the Singley and Anderson (1985) editor experiments, experiments done by Elio (1986) and Frensch (1991) in which subjects solved complex arithmetic problems, and experiments by Chein and Morrison (2010) and Karbach and Kray (2009) in which training on one task of executive control improved performance on others.

Modelling Working Memory

Klaus Oberauer (k.oberauer@psychologie.uzh.ch)

University of Zürich

Working memory is the blackboard of thinking. Its limited capacity is closely related to measures of reasoning ability. Current theories of working memory are primarily verbal descriptions of mechanisms and their interactions, which are often vague and ambiguous, making it difficult to figure out what exactly they predict. I will present a new model, SOB-CS, that accounts for a broad range of experimental findings from the so-called complex-span paradigm. The complex-span paradigm is the most frequently used paradigm in cognitive psychology for investigating working memory. In complex-span tests, participants try to remember a list in correct order, and in between presentation of list items, they have to carry out brief distracting tasks, such as reading aloud words or solving arithmetic problems. SOB-CS is a connectionist model that uses distributed representations of items, their list positions, and of the material of the distractor task. Items are retained in correct order by associating each item to its list position. Memory capacity is limited because of interference by superposition of distributed representations. Distractor-task material is also obligatorily encoded, thereby adding to interference. Free time can be used to gradually remove distractors from memory, thereby reducing interference. I will present applications of SOB-CS to a number of experiments from the literature, and new experiments that test some of the key assumptions of the model.

The Emerging Toolbox of Cognitive Engineering Models

Alex Kirlik (kirlik@illinois.edu)

University of Illinois at Urbana-Champaign

In his seminal “Skills, Rules, and Knowledge” paper nearly 30 years ago, cognitive engineering pioneer Jens Rasmussen wrote: “*we do not need a single integrated quantitative model of human performance, but rather an overall qualitative model which allows us to match categories of performance to types of situations. In addition, we need a number of more detailed and preferably quantitative models which represent selected human functions and limiting properties within the categories.*” In this talk, I will illustrate in detailed fashion how contemporary cognitive engineering methodology has indeed come to exist as a toolbox of models largely as Rasmussen observed. My review and analysis is based heavily on collaborative work with my co-editor John D. Lee in developing *The Oxford Handbook of Cognitive Engineering* (in press). The presentation will include a discussion of the handbook’s sections and chapters, as well as various analyses of the entire text considered as a corpus of data. These include topic analysis, hierarchical cluster analysis, and network analysis. Results indicate that modeling to support cognitive engineering and human factors does not consist of one or even a few monolithic models or architectures, but instead as a highly diverse ecology of techniques each tailored to a particular niche in human-technology interaction.

Tutorials

Developing CLARION-based Agents with the New CLARION Library

Michael Lynch (lynchm2@rpi.edu)

Department of Language, Literature and Communication Rensselaer Polytechnic Institute, 110 Eighth Street,
Troy, NY 12180 USA
Telephone: (518) 276-3243 | Fax: (518) 276-4092

Nicholas Wilson (wilson3@rpi.edu)

Ron Sun (rsun@rpi.edu)

Cognitive Science Department Rensselaer Polytechnic Institute, Troy, NY 12180 USA

Previous tutorials on CLARION have focused mainly on presenting detailed introductions to the core theoretical concepts underlying the CLARION cognitive architecture. For this tutorial, in addition to providing a detailed introduction to the theory, we will also focus on giving participants hands-on experience using the new implementation of CLARION -- the CLARION Library, version 6.1 (written in C#). To that end, we will introduce guidelines for setting up and using basic and intermediate aspects of the library (with detailed walk-throughs for several simulation examples) as well as present several significant new features and enhancements.

As CLARION is implemented in C#, participants will learn how they can employ the CLARION library on different operating systems using either the Visual Studio or Mono development environments. By the conclusion of this tutorial, participants should be equipped with the necessary foundation to begin developing CLARION-based agents for their own applications.

Tutorial Outline

A General Overview of CLARION (15 min.)

In this section, an introduction to cognitive architectures in general, and CLARION in particular, will be presented. CLARION will be compared to various other architectures and a brief discussion of some past and current applications of CLARION will be presented along with cognitive justifications and implications.

CLARION is a unified, comprehensive theory of the mind based on two basic theoretical assumptions: representational differences and learning differences of two different types of knowledge --- implicit vs. explicit, among other essential assumptions and hypotheses.

In addition to these theoretical assumptions, CLARION is a cognitive architecture composed of four main subsystems: the Action-Centered Subsystem, the Non-Action-Centered Subsystem, the Motivational Subsystem, and the Meta-Cognitive Subsystem.

Action-Centered Subsystem Basics (30 min.)

In this section, some basic concepts of the Action-

Centered Subsystem (ACS) will be presented. The structure and design of various aspects of the ACS, along with the learning mechanisms and the properties of the model, will be presented.

The Action-Centered Subsystem is used mainly for action decision-making. In the ACS, the top level generally contains simple "State→Action" rules, while the bottom level uses multi-layer perceptrons to associate states and actions. Reinforcement learning algorithms (usually with backpropagation) are used in the bottom level while rule learning in the top level is mostly "one-shot" and can be performed bottom-up (via "explicitation") or independently (e.g., through linguistic acquisition). This section will focus on the representation for the top and bottom levels, and will detail bottom level learning and bottom-up rule extraction and refinement (RER).

Setting up and Using the ACS (30 min.)

For the first hands-on section of the tutorial, participants will be instructed on how to set up and install the CLARION Library and are walked through a simple simulation example. In addition, several core principles necessary for interacting with the library will be outlined.

Working Memory and Goals (15 min.)

In this section we will discuss the theoretical underpinnings for the working memory (WM) and the role that goals play in the decision-making processes of the ACS.

The working memory is conceived as a requisite structure within the ACS, whereas goals are stored within a top-level construct of the Motivational Subsystem, referred to as the Goal Structure (GS).

Setting up and Using the WM and GS (15 min.)

For this hands-on section, participants will be shown both the manual and action-oriented methods for setting-up and using the working memory and goal structure. In addition, a simple simulation example will be presented that demonstrates the use of working memory.

Drives and Meta-Cognitive Modules (30 min.)

This section will focus on the structure and design of the motivational (MS) and meta-cognitive (MCS) subsystems. In particular, the drives within the MS and various meta-cognitive modules within

the MCS, will be described.

The Motivational Subsystem contains both low-level (physiological) and high-level (social) primary drives that take into account both environmental and internal factors in determining drive strengths. These drive strengths are reported to the Meta-Cognitive Subsystem, which regulates not only goal structures but other cognitive processes as well (e.g., monitoring, parameter setting, etc).

Setting up and Using Drives and Modules (30 min.)

For this hands-on section, participants will be shown both the manual and action-oriented methods for setting up and using the working memory and goal structure. In addition, a simple simulation example will be presented that demonstrates the use of these mechanisms.

Hands-On Practice Session #1 (15 min.)

In the final section before lunch, participants will be given the opportunity to further explore the CLARION Library and the simulations that were presented to this point. Participants will also be encouraged to ask any questions they may have with regard to using the library at this time.

The Non-Action-Centered Subsystem (45 min.)

Similar to the section on the ACS, this section will detail the Non-Action-Centered Subsystem (NACS). The structure and design of the various aspects of the NACS, along with the learning mechanisms and the theorems describing the properties of the model, will be presented.

The Non-Action-Centered Subsystem stores declarative (“semantic”) and episodic knowledge and is responsible for reasoning in CLARION. In the NACS, the top level contains simple associations while the bottom level involves nonlinear neural networks. Associative learning algorithms (e.g., backpropagation or contrastive Hebbian) are generally used in the bottom level whereas associations in the top level are mostly learned “one-shot” (similar to the ACS).

Performing Reasoning using the NACS (15 min.)

For this hands-on section, participants will be given a very brief introduction to using the reasoning mechanism in the NACS. However, as the NACS is currently in the development stage, this demonstration will necessarily be brief.

Intermediate Aspects of the ACS (30 min.)

In this section we will discuss several intermediate concepts for the ACS. In particular, we will review the theoretical considerations that govern IRL and Fixed rules.

IRL and Fixed rules are the other two forms of procedural knowledge (besides RER rules) that can be found in the top level of the ACS.

Setting-up and Using IRL and Fixed Rules (30 minutes)

For this hands-on section, participants will be

shown how to do some basic customization using the CLARION Library. In particular, we will show participants how to use C#'s *delegate* concept in order to quickly and easily create their own customized rules. In addition, a simple simulation that uses IRL rules will be presented.

Pre-Training, Tuning and Parameter Setting (15 minutes)

For this hands-on section, participants will be shown several methods for performing simple tuning and parameter setting operations in the CLARION Library.

Features and Plugins (15 minutes)

For this hands-on section, participants will be shown some of the useful features and plugins that are currently available as part of the CLARION Library.

Hands-On Practice Session #2 (30 min.)

In the final section of the day, participants will be given the opportunity to further explore the CLARION Library and ask any additional questions they may have.

Relevance for Cognitive Science

The CLARION cognitive architecture is well established with over 100 scientific papers and several books. CLARION is particularly relevant to cognitive scientists because of its strong psychological plausibility and the breadth of its application to cognitive modeling and simulation. In CLARION, each structure corresponds to a psychological process/capacity. CLARION-based models have been used to explain data as diverse as implicit learning, cognitive skill acquisition, inductive and deductive reasoning, meta-cognition, motivation, personality, and social simulations.

Presentation Details

Descriptions and demonstrations during the presentation will be provided using PowerPoint and the Visual Studio and Mono development environments.

Participants in the tutorial are encouraged to ask questions throughout the presentation to clarify any ideas described.

Sample Materials

- Sample slides:
<https://sites.google.com/site/clarioncognitivearchitecture/presentations>
- A complete technical specification of CLARION:
<http://www.cogsci.rpi.edu/~rsun/sun.tutorial.pdf>
- A list of CLARION-related publications:
<http://www.cogsci.rpi.edu/~rsun/clarion-pub.html>
- The current (6.1.0.6, C#) and previous (6.0.5, Java) versions of the CLARION Library:
<https://sites.google.com/site/clarioncognitivearchitecture/downloads>
- Other demonstration materials: See the "Tutorials" folder within the current CLARION Library software package

Scaling models of cognition to the real world: Complexity-theoretic tools for dealing with intractability

Iris van Rooij (i.vanrooij@donders.ru.nl)
Donders Institute for Brain, Cognition, and Behaviour
Radboud University Nijmegen

Johan Kwisthout (kwisthou@liacs.nl)
Leiden Institute of Advanced Computer Science
Leiden University

Introduction

A common theoretical obstacle encountered by computational- or rational-level models of cognition is that the cognitive capacities that they postulate appear to be computationally intractable (e.g., NP-hard or worse). Formally, this means that the computations that these models postulate consume an exponential amount of time. Informally, this means that the postulated computations do not scale in any obvious way to explain how cognitive capacities can operate in the real world outside the lab. How can cognitive scientists overcome this undesirable property of models of cognition? Over the last decade, several sophisticated complexity-theoretic techniques have been developed in theoretical computer science that can be utilized by cognitive modelers to systematically generate hypotheses about model changes or constraints that yield computational tractability without loss of the general applicability of the models. With this workshop we aim to bring these complexity-theoretic techniques to the attention of a broad audience of cognitive modelers and illustrate how they can be used to make cognitive models that scale to situations of real-world complexity.

Morning Session

In the morning session the tutorial organizers, Van Rooij and Kwisthout, will give a conceptual primer on computational complexity analysis in the context of cognitive modeling. The session will include a conceptual introduction to tractable cognitive modeling. Subsequently, they will review complexity-theoretic concepts (e.g., NP-hard, fixed-parameter tractability) and techniques (e.g., polynomial-time and parameterized reduction). Participants will have opportunity to practice the techniques via hands-on exercises (these can be done using paper and pencil). Also more controversial issues will be topic of discussion, such as the question to what extent intractable computations can be efficiently approximated by randomized or heuristic methods. The organizers aim for an interactive style of discussion.

Reading material: van Rooij, I. (2008). The Tractable Cognition thesis. *Cognitive Science*, 32, 939-984.

Afternoon session

In the afternoon session, four speakers will illustrate several applications of the concepts and techniques introduced in the morning session. Each application talk will consider a different type of model in a different cognitive domain.

What does (and doesn't) make deriving analogies hard?

Todd Wareham (Memorial University of Newfoundland) will present complexity analyses of Structure-Mapping Theory (SMT), assessing several conjectures in the literature about conditions that make analogy derivation under SMT feasible in practice.

Does recipient design make intention recognition tractable?

Mark Blokpoel (Radboud University) will consider Bayesian models of intention recognition and recipient design in the context of communication. He will demonstrate these models are NP-hard but also identify model constraints that yield computational tractability.

A tractability border in natural language semantics

Jakub Szymanik (University of Groningen) will discuss how ambiguity in natural language may be related to computational complexity. He will focus on logic-based models of quantifier expressions (e.g. 'some', 'more than') and will outline a tractability border between quantifier sentences.

Is managing multiple goals an intractable balancing act?

Daniel Reichman (Weizmann Institute of Science) will put forth the idea that people find it difficult to achieve multiple goals simultaneously because doing so entails solving computational intractable problems. He will outline approaches that can aid people in solving hard problems related to the attainment of multiple interrelated goals.

For more information about this tutorial, full details of the schedule, and extra materials, please refer to our website: <http://tcs.dcc.ru.nl/iccm2012/>

Design principles revisited: The continued design of the Symbolic and Sub-symbolic Robotics Intelligence Control System (SS-RICS)

Troy Dale Kelley (troy.d.kelley6.civ@mail.mil)

U.S. Army Research Laboratory Human Research and Engineering Directorate Aberdeen Proving Ground,
Aberdeen MD 21005

Abstract

The Symbolic and Sub-symbolic Robotics Intelligence Control System (SS-RICS) is a production system based robotics controller based largely on the cognitive architecture the Adaptive Character of Thought-Rational (ACT-R). At the beginning of the research program a set of design principles were developed to aid in the design of the robotics system. These principles are discussed and revisited here.

Keywords: cognitive architectures, cognitive modeling, robotics

Introduction

In the last several decades, cognitive architectures have been designed around psychological principles in an attempt to reproduce the thought patterns of the human mind (Anderson & Lebiere, 1998). These cognitive architectures have made progress in modeling the human mind by using the production system architecture as a basis; however, they traditionally have had little interaction with the outside world which gives them limited functionality as real-world robotics controllers. The Sub-symbolic Robotic Intelligence Control System (SS-RICS) was developed using a production system as the central executive, as with traditional cognitive architectures, while also using sub-symbolic algorithms for perceptual processing. This allows SS-RICS to interact with the outside world. Additionally, these perceptual sub-symbolic algorithms are run in parallel with the production system, and mimic the parallel perceptual processing seen in the humans and animals. Additionally, the production system within SS-RICS is capable of shutting down certain algorithms (i.e. face recognition) if the current goal does not require the specified algorithms, thereby freeing up computational resources.

SS-RICS is part of an ongoing development within the U.S. Army Research Laboratory of a robotic control architecture that was inspired by computational cognitive architectures, primarily the Adaptive Control of Thought – Rational (ACT-R). SS-RICS combines symbolic and sub-symbolic representations of knowledge into a robotic control structure that allows robotic behaviors to be programmed in a production system format. The architecture is organized as a goal driven, serially executing, production system at the highest symbolic level; and a multiple algorithm, parallel executing, simple collection of algorithms at the lowest sub-symbolic level.

Five Development Principles

In order to guide the development of SS-RICS, five development principles were established in 2009 (Kelley et al. 2009).

- 1) The lowest level of perception includes algorithms running in a parallel fashion, while the highest levels of cognition are algorithms operating in serial fashion
- 2) At both the low levels and the high levels of cognition, the algorithms are relatively simple. It is the interaction, processing and results of simple algorithms which produce complex intelligent behavior.
- 3) Pre-programming SS-RICS is guided by the algorithms that are recognized as part of the human evolutionary process (for example, algorithms for edge detection, auto-focus of the eyes, pupil dilatation in different lighting environments). The pre-programming that is done should allow for the emergence of complex behavior, but not be the complex behavior itself.
- 4) Cognitive development within SS-RICS is principally about the reorganization of memory elements through increasing and decreasing their respective strengths.
- 5) Cognitive development and change can occur after allowing for specialized internal processing (i.e. dreaming) or after the necessary low level elements (i.e. features) are in place to allow for higher level symbolic extraction.

Developmental principles revisited

As defined in principle one, we have found an enormous value in running perceptual algorithms (motion tracking) in parallel with our other sub-symbolic algorithms (finding corners or gaps in a wall). This allows the higher levels of the system to turn off perceptual algorithms as the system becomes overloaded or runs out of memory; or allows us to pick and choose what functionality we are interested in, depending on the task. This can make the system very adaptive to certain tasks and make it able to use all of the available processing power for a given task. Additionally, we are currently running the cognitive process in serial but have found some utility in running multiple cognitive processes in parallel. In other words, the algorithm for the identification of an object is running in parallel with the algorithm for the identification of a specific face. The reader might

ask – “what is the line between cognitive processes and perceptual processes” and it should be noted that this distinction can sometimes become blurred. It is not entirely clear that the identification of a face is, in fact, a cognitive process. We would rather reserve cognitive processes to strategy selection and problem solving so these lower levels processes should be, perhaps, pushed down to the parallel aspects of perception. This would make our goal stack relatively simple and would make the production system relatively simple to program.

As outlined in principle two, our algorithms, in general, remain relatively simple, except in some cases where we are using traditional AI techniques like Principle Component Analysis (PCA) or algorithms involved with Simultaneous Localization and Mapping (SLAM) (i.e. particle filters). While we strive to use cognitively plausible algorithms, traditional robotics algorithms can be seen as a means to an end for certain behaviors. For example, it is useful to use some SLAM algorithms to allow the robot to move from one room to the next, while more cognitively based algorithms like spreading activation can be used for object identification along the way.

Pre-programming algorithms based on evolutionary processes continues, and we feel we have adhered to principle three. However, when one considers the number algorithms humans are endowed with through evolution (i.e. color identification, sound localization, pupil dilatation based on light levels, object identification, object tracking, movement identification, contrast illumination.. and so forth), this can be a daunting task. Indeed, we have found this to be one of the more difficult and time consuming aspects of implementing an intelligent robotics system. It is important for any robotics engineer to realize that many of these low level algorithms need to be in place before any more complex behavior can emerge from an intelligent system. And while many of these algorithms seem intuitively simple (object identification)

their implementation and interaction with other algorithms can create challenging developmental issues.

The reorganization of information as outlined in principles four and five continues to be an issue. We have not used proceduralization as implemented within ACT-R and would like to use this process to reduce the number and size of the goals developed by programmers. The struggle to write simple and powerful goals continues to be an issue, and we have looked at using subsumptive architectures to reduce the number and size of the goals. However, as I have pointed out in other articles, you cannot simulate extremely complex behaviors (i.e. playing chess) with a subsumptive architecture (Kelley and Long, 2010), and more powerful planning and strategy selection behaviors must still be written by hand or generated by some relatively complex process.

The abstraction and generalization of memories, as

outlined in principle five, especially different types of memories (declarative, procedural and episodic) continues to be an area of continued research within SS-RICS. Interestingly, we have found some computational support

for the concept of off-line processing or dreaming based on the speed of different memory retrievals. As part of our development of SS-RICS we found that real time retrieving memories for moving objects slows the system down too much, and it is better to try and remember everything that happens and consolidate these memories in order to speed retrievals. During consolidation, an off-line strategy to activate important memories is used and subsequent retrieval times can be greatly increased. Specifically, by increasing the strength of important memories using bottom-up activation, certain perceptions can then be selected by the cognitive system depending on their task relevance. This is more efficient than trying to identify everything that happens in real time. This would be an evolutionary argument for dreaming, which consolidates memories and speeds their retrieval times for the efficient execution of future recognitions.

Conclusions

SS-RICS continues to be undergo a complex and challenging development cycle, where new developments occur each day. We feel we have adhered to our original design guidelines and will continue to use these guidelines to further the development of the system.

References

- Anderson, J. R., and Lebiere, C. *The Atomic Components of Thought*: Erlbaum, 1998.
- Kelley, T.D., Avery, E., Long L.N., & Dimperio, E. (2009). A Hybrid Symbolic and Subsymbolic Intelligent System for Mobile Robots. Presented at the AAIA Infotech@Aerospace Conference, AAIA-2009-1949, Seattle, WA, April 6-9.
- Kelley, T. D., & Long, L.N. (2010). Deep Blue cannot play checkers: The need for generalized intelligence for mobile robots. *Journal of Robotics*, 2010, 1-8.

Tutorial: Understanding cognitive processes through language use

Thora Tenbrink (tenbrink@uni-bremen.de)

SFB/TR 8 Spatial Cognition

University of Bremen, Germany

<http://www.informatik.uni-bremen.de/~tenbrink/>

Keywords: Problem solving; process models; linguistic structure; think-aloud data; retrospective reports; complex cognition

Introduction

How can cognitive processes be accessed and understood sufficiently to enable reliable computational models? One established way of addressing internal processes is to analyze their external representations, most prominently natural language produced along with cognitively complex tasks (Ericsson & Simon, 1993). The aim of this tutorial is to familiarize both young and experienced researchers with the systematic and linguistically informed analysis of language data collected in order to substantiate cognitive models. The method of *Cognitive Discourse Analysis* (CODA) (Tenbrink & Gralla, 2009; Tenbrink, 2010) will be introduced, which uses linguistic methods and insights to address research questions in cognitive science. One main aim is to identify particular types of linguistic patterns in the collected data that are likely to point to specific cognitive processes. The outcome of a CODA-based analysis is a validated account of systematic cognitive processes feeding directly into subsequent computational cognitive modelling.

Methods that employ language to address research questions in cognitive science range from psychological via psycholinguistic approaches to linguistic discourse analysis. In spite of their fundamental diversity, such methods share the basic view that patterns in language are systematically related to patterns of thought (Chafe, 1998). A prominent feature and aim of the CODA framework is to identify relevant types of linguistic patterns that are likely to point to specific cognitive processes in diverse scenarios. Systematic accounts of recurring patterns of thought and prominent conceptualizations provide a substantial prerequisite for cognitive modelling approaches of any kind.

CODA can be employed to enhance the analysis of think-aloud protocols and retrospective reports for the identification of (internal) *cognitive processes* (Ericsson and Simon, 1993; Tenbrink, 2008). Conventionally, the focus in this kind of analysis lies on the *content* of verbal data, addressing those aspects (e.g., particular thought processes or strategies) that the speakers are themselves aware of. The content-based inspection of verbal reports, particularly if carried out by experts in the problem domain and set against a substantial theoretical background (Krippendorff, 2004), often leads to well-founded specific hypotheses about the cognitive processes involved. The detailed systematic analysis of linguistic features and structures in CODA provides a particularly sound basis for using the language

data as evidence (e.g., Hölscher et al., 2011; Tenbrink et al., 2011; Tenbrink & Seifert, 2011; Tenbrink & Wiener, 2009).

CODA is used to gain insights into generalizable cognitive phenomena that go beyond conscious reflection by individual speakers, and that may not necessarily be directly observable in linguistic content. Speakers may not be aware of the cognitive structures that are reflected in particular ways of framing a representation linguistically. Furthermore, they may not be consciously aware of the underlying network of options (Tenbrink & Freksa, 2009) that allows for a range of linguistic choices beside their own, which emerges more clearly by considering a larger data set collected under controlled circumstances. According to previous research in cognitive linguistics and discourse analysis (e.g., van Dijk, 2008), linguistic features such as the verbal representation of semantic domains reflected in ideational networks, lexical omissions and elaboration, presuppositions, hesitation and discourse markers, and the like all indicate certain conceptual circumstances; these are related to the current cognitive representations in ways that distinguish them from other options available in the network. In particular, the chosen linguistic options reflect what speakers perceive as sufficiently relevant to be verbalized, as well as the information status assigned to the diverse parts of the verbalization.

Besides building on established insights about the significance of particular linguistic choices, validating evidence for the relationship between patterns of language use and the associated cognitive processes can be gained by triangulation, i.e., the combination of linguistic analysis with other types of evidence such as behavioral performance data. In these combined ways, data collected in empirical studies serve as validated evidence for subsequent computational modelling of complex cognitive processes.

Format and schedule

This tutorial is designed to cover a half day (three hours). Rather than offering primarily theoretical insights, the tutorial will take the participants' current or intended projects as a starting point to address the following issues, supplemented wherever suitable by practical exercises.

Motivation: How can language data serve as empirical evidence for cognitive modelling?

Data collection: What kinds of issues need to be considered in the light of actual research purposes?

CODA based analysis (main part): Systematic data annotation and interpretation, substantiated by linguistic insights.

Triangulation and systematization: How can the insights gained from language be complemented by other types of empirical data and systematized for modelling purposes?

In contrast to previous offerings, this tutorial will focus on the systematic identification of the cognitive steps and principles that can be fed into computational models.

Target audience information

There are no particular prerequisites for attending this tutorial. It will be open for researchers in cognitive science at any point in their career, ranging from graduate students to established experts in cognitive modelling.

Linguistic knowledge or expertise is welcome but not a prerequisite for this tutorial. Participants are encouraged to bring examples of their own collected natural language data as handouts or on their computers. Sample data collected in relevant scenarios will be discussed, tailored to the participants' current focus of interest.

Acknowledgments

Funding by the DFG, SFB/TR 8 Spatial Cognition, I6-[NavTalk] is gratefully acknowledged. Further thanks go to my collaborators across diverse research projects that continue to support the development of the methodology.

References

- Chafe, W. (1998). Language and the Flow of Thought. In M. Tomasello (ed.), *The New Psychology of Language. Cognitive and Functional Approaches to Language Structure*. Mahwah: Lawrence Erlbaum.
- van Dijk, T.A. (2008). *Discourse and Context. A Sociocognitive Approach*. Cambridge: Cambridge University Press.
- Ericsson, K.A., & Simon, H.A. (1993). *Protocol analysis: Verbal reports as data*. Cambridge, MA: MIT Press.
- Hölscher, C., Tenbrink, T., & Wiener, J. (2011). Would you follow your own route description? *Cognition* 121, 228-247.
- Krippendorff, K. (2004). *Content Analysis: An Introduction to its Methodology* (2nd ed.). London: Sage.
- Tenbrink, T. (2008). The verbalization of cognitive processes: Thinking-aloud data and retrospective reports. In W. Ramm & C. Fabricius-Hansen (eds.), *Linearisation and Segmentation in Discourse. Multidisciplinary Approaches to Discourse 2008 (MAD 08)*, Feb 20-23 2008, Lysebu, Oslo. Oslo: Dept. of Literature, Area Studies and Europ. Languages, Univ. of Oslo.
- Tenbrink, T. (2010). CODA: Kognitive Diskursanalyse. In: E. Ruijndijk, T. Stolz, & J. Trabant (Hrsg.), *Linguistik im Nordwesten: Beiträge zum 1. Nordwestdeutschen Linguistischen Kolloquium*. Bochum: Brockmeyer.
- Tenbrink, T., Bergmann, E., Konieczny, L. (2011). Wayfinding and description strategies in an unfamiliar complex building. In L. Carlson, C. Hölscher, & T.F. Shipley (Eds.), *Proceedings of the 33rd Annual*

Conference of the Cognitive Science Society. Austin, TX: Cognitive Science Society.

Tenbrink, T., & Freksa, C. (2009). Contrast sets in spatial and temporal language. *Cognitive Processing* 10 Supplement 2, S322-S324.

Tenbrink, T., & Gralla, L. (2009). Accessing complex cognitive processes via linguistic protocol analysis. In U. Schmid, M. Ragni, M. Knauff (Eds.): *Proceedings of the KI 2009 Workshop Complex Cognition*, Paderborn, Germany, September 15, 2009.

Tenbrink, T., & Seifert, I. (2011). Conceptual Layers and Strategies in Tour Planning. *Cognitive Processing* 12:1, 109-125.

Tenbrink, T., & Wiener, J. (2009). The verbalization of multiple strategies in a variant of the traveling salesperson problem. *Cognitive Processing* 10:2, 143-161.

Papers & Abstracts

Modeling Intuitive Decision Making in ACT-R

William G. Kennedy (wkennedy@gmu.edu)

Krasnow Institute for Advanced Study, 4400 University Drive
Fairfax, VA 22030 USA

Robert E. Patterson (Robert.Patterson@wpafb.af.mil)

Air Force Research Laboratory,
Wright-Patterson AFB, OH 45433 USA

Abstract

One mode of human decision-making is considered intuitive, i.e., unconscious situational pattern recognition. Implicit statistical learning, which involves the sampling of invariances from the environment and is known to involve procedural (i.e., non-declarative) memory, has been shown to be a foundation of this mode of decision making. We present an ACT-R model of implicit learning whose implementation entailed a declarative memory-based learner of the classification of example strings of an artificial grammar. The model performed very well when compared to humans. The fact that the simulation of implicit learning could not be implemented in a straightforward way via a non-declarative memory approach, but rather required a declarative memory-based implementation, suggests that the conceptualization of procedural memory in the ACT-R framework may need to be expanded to include abstract representations of statistical regularities. Our approach to the development and testing of models in ACT-R can be used to predict the development of intuitive decision-making in humans.

Keywords: implicit learning; cognitive models; unconscious learning; ACT-R theory.

Introduction

The vast majority of cognitive models discussed at this conference are models of rational or analytical cognition. The architectures used are primarily ACT-R (R for rational) or Soar, and the best papers compare a computational implementation of a theory, i.e. a model, to human behavior observed in careful laboratory experiments. The authors of these papers then claim that the model under consideration is a plausible theory of the cognitive process behind the observed behavior. This approach is advancing the understanding of cognitive processes. However, modeling consciously rational behavior addresses only part of human cognition and it ignores the ubiquitous influence that implicit processing and intuitive decision making has on human behavior.

In the dual-process framework of reasoning and decision making (e.g., Evans, 2008; Patterson, Pierce, Bell, Andrews & Winterbottom, 2009; Sloman, 1996), one mode of decision making is called intuitive. Intuitive decision making refers to implicit situational pattern recognition that is not thought to involve symbolic rules (Klein, 1998). The other mode of decision making is called analytical, which is generally accepted to entail symbolic rules. Intuitive decision making, which falls under the rubric of 'System 1'

processing in this literature, is typically described as unconscious, fast, and effortless decision making. Analytical decision making, which falls under the rubric of 'System 2' processing, is described as conscious, rational, slow, and effortful. Evans (2008) provides a review of the evidence supporting the Dual-Process theory. Analytical decision making is relatively simple to study because it is easy to create tasks for testing and recording behavior during rational performance. Intuitive decision making, on the other hand, is difficult to study because it is hard to artificially create environmental patterns with sufficient fidelity to study situational pattern recognition.

Recently, Patterson and colleagues (Boydston, Patterson, Pierce, Park & Tripp, 2011; Covas-Smith, Patterson, Pierce, Cooke & Homa, 2011; Patterson et al., 2009) have investigated the development of intuitive decision making in a simulated real-world environment. These authors had human participants experience simulated flight over a synthetic terrain with a sequences of objects (e.g., house; vehicle) positioned on the terrain along the flight path. Each object sequence was derived from paths taken through a finite-state algorithm, which defined a grammar for constructing the content of the scene. The use of a finite-state grammar for creating object sequences was analogous to the way in which finite-state grammars have been used for studying the implicit learning of artificial letter strings (e.g., Reber, 1967). Patterson and colleagues tested the conjecture that implicit learning (Cleeremans, Destrebecqz & Boyer, 1998; Perrachet & Pacton, 2006) could be one way in which intuitive decision making is developed.

Patterson and colleagues found that naive participants could implicitly learn the object sequences quite easily. Moreover, the implicit learning of the sequences provided a foundation for intuitive decision making about the underlying structure of the sequences: following training with the artificial object sequences, the participants were successful in recognizing novel sequences taken from the same grammar during test. That is, the human participants implicitly learned to recognize situational patterns.

The ACT-R architecture (Anderson, 2007; Anderson, et al., 2004) has been used before to model implicit learning. In particular, Wallach and Lebiere (2003) reviewed the theoretical approaches to implicit learning and observed "a major shortcoming of these models is their failure to also account for explicit learning and for the difference between implicit and explicit learning" (pg 217). They then presented

ACT-R models of two well-known implicit and explicit learning tasks and specifically linked explicit learning with the learning of declarative chunks and implicit learning with ACT-R's sub-symbolic learning of the activations of those chunks. We will use the same approach here, using the sub-symbolic representation associated with declarative memory as the basis of our model of intuitive decision making.

This paper presents an approach to studying intuitive decision making (i.e., System 1 cognition) that exposes the cognitive process to computational modeling and experimental testing of theories implemented as models. An ACT-R model was developed and compared to human subject data on an intuitive decision-making task used by Patterson, Pierce, Boydston, Park, Shannan, Tripp and Bell (submitted).

Patterson et al. Study

Patterson, Pierce, Boydston, Park, Shannon, Tripp, and Bell (submitted) investigated whether implicit learning can be a process by which intuitive decision making is acquired. One form of implicit learning entails the learning of spatial or temporal patterns without full awareness of what is learned (Cleeremans, Destrebecqz & Boyer, 1998; Perrachet & Pacton, 2006). Implicit learning is likely to be a key process by which individuals learn situational patterns on which intuitive decisions are based (e.g., Patterson et al., 2009).

Patterson et al. extended the classic paradigm by Reber (1967) used for studying implicit learning, which entailed the learning of a synthetic grammar produced by a finite state algorithm that generated artificial letter strings. Patterson et al. instead investigated the implicit learning of passively viewed, structured object sequences presented in a simulated real-world immersive environment used for simulating locomotion (Figure 1). In doing so, they used a finite state algorithm that created an artificial grammar for generating the object sequences and thus the content of the environment (Figure 2). For comparison, Patterson et al. also investigated the implicit learning of *memorized* static letter strings presented on a flat display, as has been done in the past (Reber, 1967) (See Figure 3).

The finite-state diagram of the grammar shown in Figure 3 has also been used in many other studies (Cleeremans, Destrebecqz, & Boyer, 1998; Matthews, et al., 1989; Perrachet & Pacton, 2006). It produces 44 valid structured strings of length 8 or shorter. Participants are trained by being presented a series of example strings from the grammar and are then tested by being asked if a test string is legal or not.

During training, Patterson et al. had human participants (1) passively view structured sequences of objects presented on a dynamic terrain seen in perspective view (the 'immersive display' condition), or (2) memorize structured strings of letters presented on a static flat display. Following training, participants were tested for implicit learning by making intuitive pattern-recognition judgments of novel

structured object sequences or letter strings versus random sequences or strings.

By training participants on the structured object sequences or letter strings, and then testing recognition of structured versus random sequences or strings, the participants performed an '*anomaly recognition*' test. The random sequences or strings effectively served as an anomaly to be recognized because the participants were never trained on random sequences or strings.



Figure 1. Photograph showing the simulated real-world environment. The scene underwent expansive optic flow motion, which simulated passive movement by the participant in the forward direction toward the horizon.

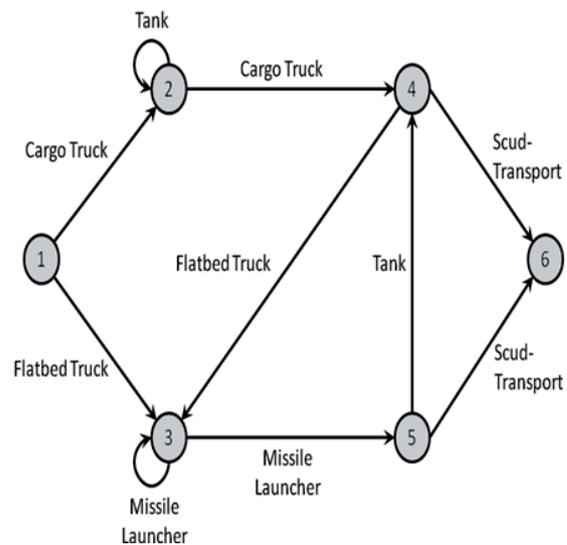


Figure 2. Depiction of finite state algorithm that defined the grammar employed for generating the structured sequences of objects used in Patterson, Pierce, Boydston, Park, Shannon, Tripp, and Bell (submitted).

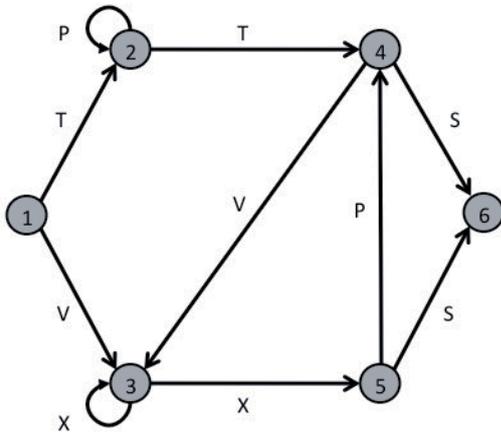


Figure 3. Finite state algorithm that defined a grammar of letter strings with the same structure as the sequences of objects of Figure 2. (From Reber, 1967.)

Results. Figure 4 depicts results obtained for the simulated real-world environment and for the static flat display, as reported by Patterson et al. Passive viewing of object sequences (third bar from the left) resulted in an average accuracy of intuitive decision making that was equivalent to the average recognition performance that was obtained when letter strings presented on a flat display were memorized (middle bar). An a-priori *t*-test showed that the difference between these two conditions was not significant, $t(14) = 0.4$, $p = 0.7$. The two training conditions were significantly higher than the no-training control condition (left bar), which was at chance-level performance.

During debriefing, the human participants had trouble explicitly verbalizing all of what they had learned during training and that a number of their decisions made during testing were from a feeling "in the gut". Thus, the training methods produced a significant level of implicit learning that was a foundation for the pattern-recognition-based (intuitive) decision making.

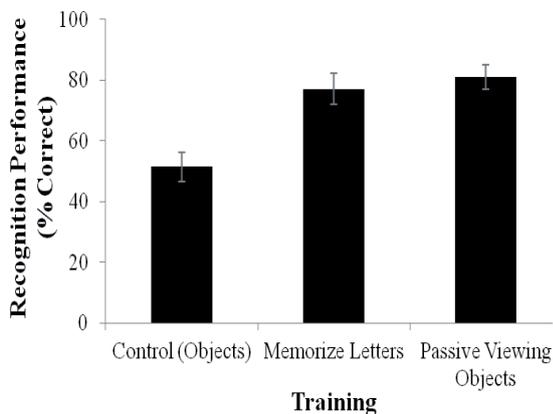


Figure 4. Accuracy of intuitive decision making, as measured by recognition performance during the test phase. During training, the object sequences were passively viewed, and the letter strings were memorized. The control

group, which involved object sequences, entailed no training. Each data point is the mean of eight human participants; error bars depict ± 1 standard error of the mean (SEM). The data was from one portion of Patterson et al. (submitted).

ACT-R Model

Our model is based on ACT-R (Anderson, 2007; Anderson, et al., 2004). ACT-R is a rule-based architecture representing cognitive processes symbolically and sub-symbolically. Its declarative memory holds chunks of declarative facts with an activation level based on the recency and frequency of use. IF-THEN rules are held in a long-term procedural memory. ACT-R models can learn by adjusting the activation of accumulated declarative chunks, by adjusting the relative measure of rules, or by combining sequential rules into new rules.

Statistical learning is sometimes modeled as the tuning of the relative measures of rules and that approach could have been used here. However, to study different strategies believed to be used in the implicit learning of abstract grammars, the model developed here uses the activation of declarative chunks of memory, with each chunk representing a bigram of letters. (The task modeled was the letter string version of the implicit learning task. An analogous model would apply to the object sequence version of the task.)

Our ACT-R model uses both the declarative and procedural modules to passively learn and then respond to this task. The rules are fixed during the run of the model. Declarative memory chunks are added based on experience during training and are recalled to make the valid/invalid evaluation during testing.

During training, rules direct the system to read the string letter by letter, left to right. The system then forms declarative chunks and saves them as an internal representation of the grammar based on observed training strings. Each declarative fact is a representation of observed bigrams indicating which letter was seen before another, i.e., a first letter and a predicted second letter.

During testing, a representation of the intuitive decision-making process determined whether all the bigrams in a test stimulus have been seen before. To respond, the system reads the string left to right and attempts to recall bigrams predicting the next letter. A successful retrieval increases the activation of that declarative chunk and the ACT-R architecture returns the one declarative chunk for an attempted recall operation. If the retrieved bigram does not match the second letter, a second retrieval is attempted using both the first and second letter. If successfully recalled, the evaluation continues. If not successfully recalled, the test string is evaluated as invalid.

This approach implements a form of predictive and evaluative behavior. Other approaches that could have been studied include recalling the first few letters of a string (primacy), recalling the last few letters of a string (recency), recalling both the first few and the last few letters, or simply

deciding by noting whether the number of pairs of letters recallable was above a threshold. We could also have tested trigram representations, or other representations. The model discussed here based its decisions on whether predictive bigrams were recallable.

Replicating Human Subject Experiments

To compare the ACT-R model to data from Patterson et al (submitted), we replicated the passive training and testing protocols for strings of letters generated by the artificial grammar shown in Figure 3. The human participants and the model were trained and tested the same way.

Training used 18 unique strings drawn from the 44 valid strings of length 8 or less. Training was organized as six blocks of three unique valid strings, which were presented 16 times with each string presented for 5 seconds and a blank screen shown for 0.6 seconds between strings.

The testing process also replicated that used with human participants. The system presented 88 strings, 22 valid strings that were not used for training, and 22 foils, each presented twice in a random order. The foils used the same letters, but in a random order and were of length 6, 7, or 8.

For this model, only default ACT-R parameters were used, except the retrieval threshold (:rt) for declarative memory and the activation noise variable (:ans).

Model's Performance

The model reports whether a test string is valid or not for each of the 88 trials. Response accuracy was our only performance measure because the response time for the human participants was not collected in the Patterson et al. study. Figure 5 shows the performance results of the model together with the human participants' test performance as reported by Patterson et al. (submitted).

The plotted results for the human participants are the means and standard error for 8 individuals as described in Patterson et al. (submitted). The plotted model results are the means and standard error for 30 runs of the model varying the retrieval threshold (:rt) parameter from -1.5 to +3.0. The noise parameter, :ans, was 0.1. The :rt parameter sets the threshold for successful retrievals from memory base on the activation level of chunks of memory. Lower values of the parameter allow the model retrieve more instances and higher values restrict retrievals to the most activated memories.

A two-tailed, equal variance t-test found that the difference in mean accuracy between the humans and the model was not significant for the 30 runs with :rt =2.0 ($t(29) = 15.74, p < 0.001$). Note, however, that the mean accuracy for the humans and model were very similar and within a few percentage points of one another for several values of :rt. Therefore, we are encouraged by the closeness of the model behavior to that of humans on this implicit learning task.

Because we are using a computational model, it is relatively easy to collect additional information on the performance of the model and compare it to human data.

Table 1 provides the human and model performance in more detail than the summary information shown in Figure 5.

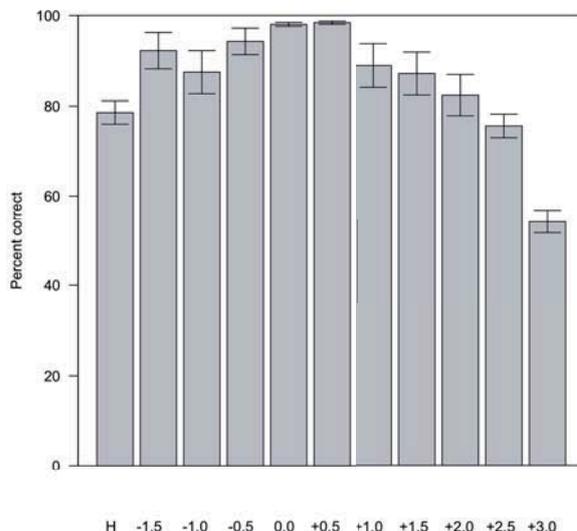


Figure 5. Model and Human Performance. The human performance (H column) and the model's performance for retrieval threshold parameter (:rt) from -1.5 to +3.0 on strings of letters learned by passive viewing. Error bars in this figure depict ± 1 standard error of the mean (SEM)

Table 1: Human and Model Performance in Detail.

Trial/Response Type	Human	Model :rt=2.0
Hits	33/44	34/44
Correct Rejections	36/44	39/44
Misses	11/44	10/44
False Alarms	8/44	5/44

In the table, "Hits" refers to the number of grammatical strings that were detected as grammatical, "Correct Rejections" refers to the number of ungrammatical strings that were detected as ungrammatical, "Misses" refers to the number of grammatical strings that were incorrectly seen as ungrammatical, and "False Alarms" refers to the number of ungrammatical strings that were incorrectly seen as grammatical. In a signal detection analysis, one can compare the hit rate and false alarm rate to get an estimate of the level of criterion that is being used for detection: a high hit rate coupled with a high false alarm rate would suggest that the detection system is overly responsive and that its actual sensitivity is not particularly high. However, a high hit rate coupled with a low false alarm rate would suggest that the system is responding selectively to a signal and that its sensitivity is high.

While the data shown in Table 1 is insufficient for a formal signal detection analysis, it is clear that both the human data and the model data reveal a very similar pattern of high hit rates coupled with low false alarm rates. This

would suggest that both systems, human and model, possess a similar high level of sensitivity for implicit learning.

Discussion

The results show that the ACT-R model can replicate the human performance when the retrieval threshold parameter is tuned to account for the training protocol. Interestingly, the model appears better at recognizing foils than hits like humans. Therefore, the results imply we have a reasonable model of the intuitive decision making process.

Intuitive decision making and its development through implicit learning depend upon a form of non-declarative memory called procedural memory. According to Squire (2004, 2009), human memory can be subdivided into two basic systems. One system is a declarative memory system, which entails conscious recollection about facts and events. The other system is a non-declarative memory system, one form of which is procedural memory, which involves memory relating to the ability to extract common elements and patterns from separate events (Knowlton, Ramus & Squire, 1992; Knowlton & Squire, 1993; 1996), as well as memory supporting the development of skill-like abilities.

Procedural memory is involved in much more than motor skill. Rather, procedural memory is also involved in the recognition of invariant properties within patterns of information that unfold over time (Patterson, et al., 2009). Because procedural knowledge is highly implicit and does not require full conscious processing to be evoked and used, it is especially useful in situations where the traditional analytical (conscious) processing of information, which is slow and limited by working memory capacity, would burden a person already stressed within a dynamic, time-pressed task environment.

ACT-R implements a formalized representation of declarative memory and non-declarative procedural memory systems and both systems have sub-symbolic components. In ACT-R, the declarative memory contains facts and events, but the retrieval of facts is not always a conscious process in that it is based on the sub-symbolic activation, which is based on the history of use of the memory.

ACT-R's procedural memory is activated by recognizing stimuli, i.e., matching the "IF" parts and then initiates actions in one or more of the architecture's modules, such as changing the current description of the goal, initiating the recall of a declarative chunk of memory, initiating a motor action, or moving the focus of the eyes. This is a different concept of "procedural" memory that discussed above.

We used ACT-R's declarative memory for facts and its sub-symbolic activation associated with those memories along with simple productions to represent the intuitive decision making process. We can produce both the overall performance as well as the different performance on hits and correct rejections implying we are modeling the cognitive processes involved. Our modeling formalization and data available raises research questions concerning the intuitive decision making process and the appropriate architectural approach. Further research will be needed to determine if

another strategy for the learning and use of the learned knowledge would also perform well compared with human data.

Modeling the non-declarative knowledge that was investigated in the present study is a challenge because this kind of procedural knowledge is more abstract than ACT-R's simple symbolic chunks, their activations, or productions, yet it is not declarative. This means that the conceptualization of non-declarative procedural memory in ACT-R may need to be expanded to include abstract representations of statistical regularities and invariances sampled from the environment.

Our ability to match available human performance data does not mean that we have proven the ACT-R model is necessarily an explanation of the underlying human cognitive processes. Humans can make the translation of their learning in one environment to another, as demonstrated by Patterson, et al. Their participants could learn pattern independent of the specific items in the sequence. However, the ACT-R model is not able to do that because the declarative chunks learned are specific to the letters in the stimuli and the knowledge are not generalizable. Further work in this area may justify extending ACT-R to represent implicit patterns more abstractly.

Conclusions

This work demonstrates some of the reasons for building computational cognitive models. First, we are able to replicate human performance on this implicit learning and intuitive decision making task. This was accomplished by implementing a model of a cognitive learning and evaluation process that, while consistent with the ACT-R theory of cognition (Anderson, 2007), was inconsistent with the intuitive nature of procedural memory in humans. The strategy implemented was to build a memory of bigrams of sequential letters and then evaluating a test string by checking that each bigram had been seen before. However, other strategies may be similarly successful.

Second, cognitive modeling supports formally exploring alternative explanations for observed behavior. The model could be modified to test whether learning trigrams in the training strings could yield similar results. It could also be modified to test if recognizing only the first few and/or the last few letters, i.e., primacy or recency, can match the human participants' performance. A third alternate strategy is simply a voting strategy where recognized bigrams are counted and if above a threshold, the model would report a match. These strategies have not yet been tested, but with a cognitive modeling environment, they can be.

Third, this work also demonstrates that at least some System 1 as well as System 2 forms of cognition can be replicated within the current ACT-R architecture, but not necessarily all. This demonstration included implicit learning and intuitive decision making. From the work of Patterson et al. (submitted), there is data on the performance of human participants who memorize training strings rather

than passively viewing them. This may be a nice example of an effortful System 2 learning strategy rather than the far less effortful System 1 passive learning strategy. The characteristics of each system need more study. To address other examples of System 1 cognition, ACT-R may need to be extended to include introspective factors representing emotional aspects of cognition such as current arousal, general mood, and temperament.

Finally, cognitive modeling advances our understanding of cognitive processes by providing a framework to represent and explore the explanation of behaviors, such as intuitive decision making, that seem to be driven by cognition that is “beyond rational”.

Acknowledgments

This work was supported in part by AFOSR/AFRL grant FA9550-10-1-0385 and the George Mason University Center of Excellence in Neuroergonomics, Technology, and Cognition (CENTEC).

References

- Anderson, J.R. (1996). Implicit memory and metacognition: Why is the glass half-full? In L. M. Reder (Ed.) *Implicit memory and metacognition*, 123-136. Mahwah, NJ: Erlbaum.
- Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* Oxford University Press.
- Anderson, J. R. & Reder, L. M. (1999). The fan effect: New results and new theories. *Journal of Experimental Psychology: General*, 128, 186-197.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglas, S., Lebiere, C., and Qin, Y. (2004). An integrated theory of mind. *Psychological Review*, 111, 1036-1060.
- Boydston, A.S., Patterson, R.E., Pierce, B.J., Park, L.M., & Tripp, L.M. (2011). Articulatory suppression affects situational pattern recognition in immersive environments. *Proceedings of the 10th Annual Conference on Naturalistic Decision Making* (pp. 157-159) Orlando, FL.
- Covas-Smith, C.M., Patterson, R., Pierce, B.J., Cooke, N., & Homa, D. (2011). Diversity of experience and intuitive decision making in an immersive environment. *Proceedings of the 10th Annual Conference on Naturalistic Decision Making* (pp. 53-156). Orlando, FL.
- Cleeremans, A., Destrebecqz, A., and Boyer, M. (1998) Implicit learning: News from the front. *Trends in Cognitive Sciences*, 2, 406-416.
- Evans, J. S. B. T. (2008). Dual-processing accounts of reasoning, judgment and social cognition. *Annual Review of Psychology*, 59, 255-278.
- Klein, G. (1998). *Sources of power: How people make decisions*. Cambridge, MA: MIT Press.
- Knowlton, B.J., Ramus, S.J. & Squire, L.R. (1992). Intact artificial grammar learning in amnesia: Dissociation of classification learning and explicit memory for specific instances. *Psychological Science*, 3, 172-179.
- Knowlton, B.J. & Squire, L.R. (1996). Artificial grammar learning depends on implicit acquisition of both abstract and exemplar-specific information. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22, 169-181.
- Knowlton, B.J. & Squire, L.R. (1993). The learning of categories: Parallel brain systems for item memory and category knowledge. *Science*, 262, 1747-1749.
- Lebiere, C. & Wallach, D. (2001). Sequence Learning in the ACT-R Cognitive Architecture: Empirical Analysis of a Hybrid Model. In R. Sun & C. L. Gilles (Eds.). *Sequence Learning: Paradigms, Algorithms, and Applications* (pp. 188-212). Berlin: Springer Lecture Notes in Computer Science.
- Lebiere, C. & Wallach, D. (2002). Explicit or implicit learning? An integrative theory of sequence learning. In *Proceedings of the 43rd conference of the German Psychological Association*.
- Matthews, R.C., Buss, R.R., Stanley, W.B., Blanchard-Fields, F., Cho, J.R., and Druhan, B. (1989) Role of implicit and explicit processes in learning from examples: A synergistic effect. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 15, 1083-1100.
- Patterson, R., Pierce, B.J., Bell, H., Andrews, D. & Winterbottom, M. (2009). Training robust decision making in immersive environments. *Journal of Cognitive Engineering and Decision Making*, 3, 331-361.
- Patterson, R.E., Pierce, B.J., Boydston, A.S., Park, L., Shannan, J., Tripp, L., and Bell, H. Implicit statistical learning creates intuitive decision making in a simulated real-world environment. *Journal of Cognitive Engineering and Decision Making*, submitted.
- Perrachet, P. and Pacton, S. (2006) Implicit learning and statistical learning: One phenomenon, two approaches. *Trends in Cognitive Sciences*, 10, 233-238.
- Reber, A.S. (1967). Implicit learning of artificial grammars. *Journal of Verbal Learning and Verbal Behavior*, 6, 855-863.
- Sloman, S. A. (1996). The empirical case for two systems of reasoning. *Psychological Bulletin*, 119, 3-22.
- Squire, L.R. (2004). Memory systems of the brain: A brief history and current perspective. *Neurobiology of Learning and Memory*, 82, 171-177.
- Squire, L.R. (2009). Memory and brain systems: 1969 – 2009. *The Journal of Neuroscience*, 29, 12711-12716.
- Wallach, D. and Lebiere, C. (2003) Implicit and explicit learning in a unified architecture of cognition. In L. Jimenez (Ed.) *Advances in Consciousness Research*, 215-250. Amsterdam: John Benjamins Publishing Company.

Dynamic Use of Multiple Analogies in the AMBR Model Causing Re-Representation of the Target

Georgi Petkov (gpetkov@cogs.nbu.bg)
Ivan Vankov (i.i.vankov@cogs.nbu.bg)
Boicho Kokinov (bkokinov@nbu.bg)

Central and East European Center for Cognitive Science, Department of Cognitive Science and Psychology,
New Bulgarian University, 21 Montevideo Street
Sofia 1618, Bulgaria

Abstract

This paper describes how the AMBR model explains multiple analogies and more specifically how the use of a superficially similar analogical base, that turns out to be inappropriate (we call it a bridge analogy), may actually lead to the re-representation of the target and the activation of a more appropriate remote analogical source. A simulation is described that demonstrates this capability of the model. A specific prediction of the model about the re-representation that the presence of the bridge analogical source is causing is tested in a psychological experiment.

Introduction

Analogy-making is considered to be a basic cognitive process that underlies much of human cognition (Hofstadter, 2001; Holyoak, Gentner, Kokinov, 2001). That is why a lot of efforts have been put to investigate this fundamental cognitive ability (Holyoak, Gentner, Kokinov, 1998; Gentner, Holyoak, Kokinov, 2001; Kokinov, Holyoak, Gentner, 2009).

Most of this research, however, is devoted to understanding single analogies, i.e. analogies between a target and a single source. While this is certainly a very wide spread phenomena, *multiple analogies* (i.e. analogies between a target and multiple sources) do play an important role as well. There are two reasons for the use of multiple analogies. The first reason is that it is not always the case that in our previous experience we do have a case close enough to the target that can help us cope completely with the new situation. We can, however, combine several previous cases each of which partially maps to the target to collectively help to solve the problem. In the early days of analogy research there were some interesting studies of multiple analogies in physics (Collins & Gentner, 1987, Clement, 1993), in astronomy (Gentner & Markman, 1997, Gentner, et al. 1997), in medicine (Spiro et al., 1989), in biology, archeology and philosophy (Shelly, 1998, 1999, 2003), in computer science (Burstein, 1986, 1988), in transportation (Veloso & Carbonell, 1993). There were even some initial computational models of multiple analogies that were trying to explain how the information from different sources is being integrated – CARL (Burstein, 1986, 1988) and a special version of the Multiple Constraint Theory (Holyoak & Thagard, 1989) suggested by Shelly (1999). However, later on the mainstream research in the field of

analogy has concentrated on the single analogy case (Gentner, 1983, 1989, Falkenheimer et al., 1989, Holyoak & Thagard, 1989, Hummel & Holyoak, 1997, Kokinov & Petrov, 2001).

This paper is returning us to the study of multiple analogies from a new perspective following the second reason to use multiple analogies: the first analogy that comes to our mind is not necessarily the best one and we may reject it and search for a better one. Thus this first analogy may play the role of facilitator that invites the second one. Some call it “*bridging analogy*”. We are interested in the dynamics of the re-representation processes that such bridging analogies trigger and how they facilitate the multiple analogies production.

The concept of bridging analogies was first introduced by John Clement and then used by Stella Vosniadou and others (Clement, 1993, 2009, Vamvakoussi, & Vosniadou, in press, Vosniadou & Skopeliti, in press). The idea is that the teacher can provide an intermediate analogical base that will be in-between the target and the desired remote analogical source. They have experimentally shown that children, students and even experts make the desired remote analogy easier if there is such a bridging analogy provided by the teacher of physics or mathematics.

In contrast, we are interested in the mechanisms of *spontaneously self-generating* of such *bridging analogies* and what their effect could be on the *re-representation of the target* and subsequent *search for better analogies*. The next section describes a simulation experiment which demonstrates the capability of the AMBR model to spontaneously come up with bridging analogies and use them in further search of a better remote analogy. Then we present the results of a psychological experiment which tests what are the influences of this bridging analogy on the evaluation of the desired remote analogy.

Simulation

The AMBR Model

We have used the AMBR model for simulating the process of spontaneous multiple analogy-making including the generation of bridging and remote analogical sources. The general AMBR model is described elsewhere (Kokinov, 1994, Kokinov & Petrov, 2000, 2001) and for the lack of space it will not be presented here again. Crucial features of AMBR are the *decentralised representation* of episodes

which allows for context-sensitive construction of the episode descriptions (past episodes are not stable static structures but are dynamically constructed on the fly); the *continuous change of the relevance* of the various representational elements which allows for dynamic processes of representation building and re-representation; the *emergent computation processes* which are based on local information processing only and depend on the computed relevance of the memory elements which allows for exhibiting context-sensitive computation.

In previous work we have demonstrated how perception and analogy-making interact in AMBR thus allowing for dynamic re-representation of ambiguous input stimuli under the pressure of the analogy-making process (Kokinov, Bliznashki, Kosev, Hristova, 2007; Kokinov, Vankov, Bliznashki, 2009). In the following simulation we are exploring AMBR's capability to produce several analogies one after another and exhibit dynamic re-representation of the target as result of these intermediate analogies.

Overview of the Simulation

The goal of the system is to find an appropriate remote analogy for the case of "a suicidal terrorist act, made by a single terrorist"; and if possible, to transfer additional knowledge or even a proposal for how to prevent further similar acts. One superficially similar potential base is the suicidal act of a kamikaze during the World War II. We expect the system easily to activate this base and to launch the analogy. However, this analogy is not good and will fail later on. The reason is that one vivid aspect of the kamikaze is their motivation: the kamikaze is typically coming from a wealth family; they are proud of their origin and culture, of their country; they perform their suicide act with pride and for the prosperity and safety of their country.

Once activated, the motivational aspect of the kamikaze situations will try to map with its analog in the terrorist situation. Thus, the question about the deep psychological motivation of the terrorist's act will "cross the mind", i.e. the system will activate it.

However, the encoded knowledge about the terrorist's motivation is that he is an immigrant for several years already; and although he has good educational and relatively good professional successes, he is not happy. He has never overcome the cultural differences; the guilty that he has left his country; and the nostalgia.

Once activated, this aspect of the target situation should activate completely different base. Namely, the base of a Bulgarian emigrant in Ireland who has the same problems to adapt himself to a different culture and, as a consequence, he beats his wife. Nevertheless that this base seems quite different from the terrorist's one, we expect it to win the analogy because of the deep structural analogy according to the motivation.

The last step for the system is to make a transfer. The story for the Bulgarian emigrant in Ireland has a happy continuation. This man has found a solution and has solved his problems. Actually, he has opened a Bulgarian restaurant and a small shop for traditional Bulgarian souvenirs. Thus, from one side, he has never uprooted fully from his country and, from other side, has deserved a

respect from the Ireland people. Spreading of his traditional culture allows to the immigrant to stop beating his wife.

Dynamic of the simulation

The target situation is represented with eight instance AMBR-agents (fig. 1). Two of them stand for the terrorist himself and for the suicidal act.

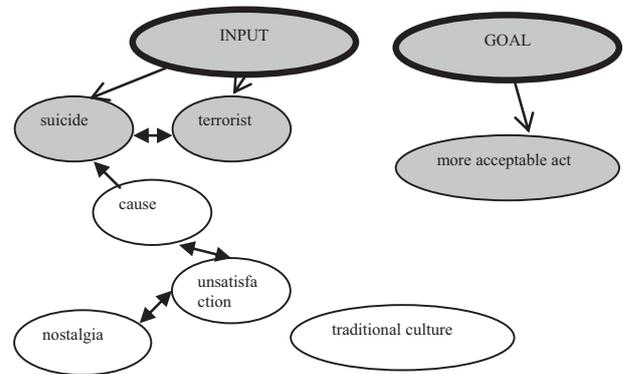


Figure 1. Schematic representation of the knowledge about the terrorist. The activation is represented with the level of gray.

These two nodes are directly attached to the INPUT and the activation spreads to the respective concepts and then back to some other known instances.

The other agents (in white on the picture) from the terrorist situations represent different aspects that the system 'knows' about the terrorists but these aspects cannot be activated easily. For example, a coalition of agents represents the deep motivation for the suicidal act of the terrorist – he is unsatisfied because of nostalgia or no acceptance of the cultural differences. However, there are not any links from the active elements to this aspect and as a consequence, the system does not 'think' about this at the beginning.

The agent 'more acceptable act' is attached to the GOAL node. Its purpose is an eventual solution to be transferred from somewhere around this agent. This agent is not connected to any other agent except its respective concept-agent.

Some other marginal pieces of knowledge are represented – for example the fact that the Arabic traditional culture is very rich and interesting for the foreigners.

One *binding-node* (not shown on fig.1), represents the whole situation. All other agents point to it, but there are few opposite links and all aspects of the situation cannot be activated from a single element.

During the first 5 AMBR cycles the activation spreads through the concepts of "suicide" and "terrorist" and then back to some typical instance. As the concept of a "japan kamikaze" is assumed to be a typical instance for a suicide, it is an opposite link from the concept of "suicide" to "kamikaze". The 'kamikaze' situation is represent again with a 'kamikaze' and 'suicide' nodes and like in the target situation the action 'suicide' is a relation with one argument

– ‘kamikaze’. This allows these pairs of nodes to be mapped easy (see Figure 2).

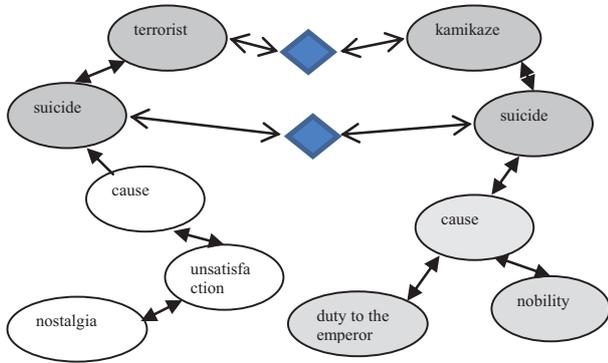


Figure 2. The mapping between the “terrorist” and “kamikaze” situations occurs at the 10th AMBR cycle. The hypothesis agents are represented with diamonds.

Once activated, the node for ‘kamikaze’ spreads activation to some other agents. The deep motivation for the kamikaze’s suicide is his honor in front of the nation, emperor and family. Thus, the activation spreads to the abstract concepts for the motivation in general, then back to the more concrete concepts and instances, and the motivational aspect of the terrorist’s act starts slowly to become active (Figure 3).

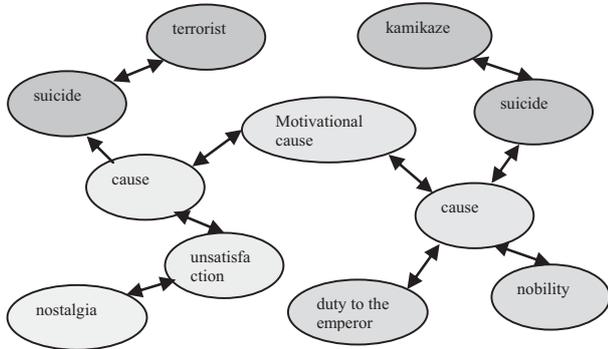


Figure 3. Between the 6th and the 19th AMBR cycles the motivational aspect of the “terrorist” story becomes active.

From other side, again from the ‘kamikaze’ base some other concepts become active because of a large number of associative links - ‘Japan’, ‘Shogun movie’, ‘England’, ‘Ireland’, etc (Figure 4).

As a result of the activation of the “immigrant” base, its elements map to the elements of the target situation. Thus, the “kamikaze” and the “immigrant” bases become competitors for the mapping with target situation.

The ‘immigrant’ base is structurally closer to the target situation, because both share the high-order relations about the motivational cause of the respective actions. Thus, nevertheless that the actions themselves are very different (the immigrant beats his wife, whereas the terrorist makes a

suicidal act), they map each other because of the pressure for structural mapping.

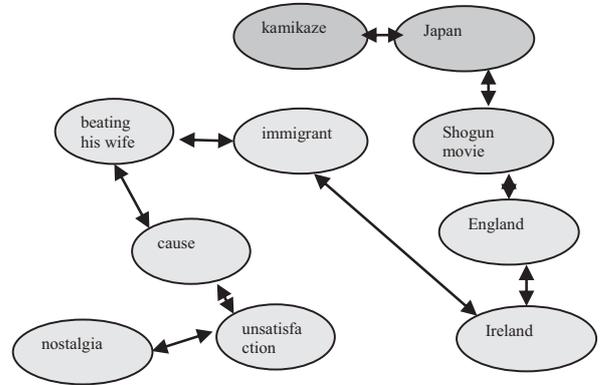


Figure 4. Between the 6th and the 17th AMBR cycles the activation spreads from the “kamikaze” base to the “immigrant” one.

Thus, at time 21 AMBR cycles (Figures 5, 6), the first mappings between the ‘terrorist’ and ‘immigrant’ situations are launched. Nevertheless, the ‘kamikaze’ situation is still more active and remains leading for a long time. The continuous structural pressure from the ‘immigrant’ situation cause firstly an inversion of the activation of the two bases (time 34); and much later the ratings are inverted too (time 77).

Finally, at time 128 the rating for the ‘immigrant’ base exceeds the threshold 1.000 and wins the competition. With other words, the hypothesis that the binding-node for the ‘terrorist’ situation corresponds to the respective binding-node for the ‘immigrant’ situation becomes a winner.

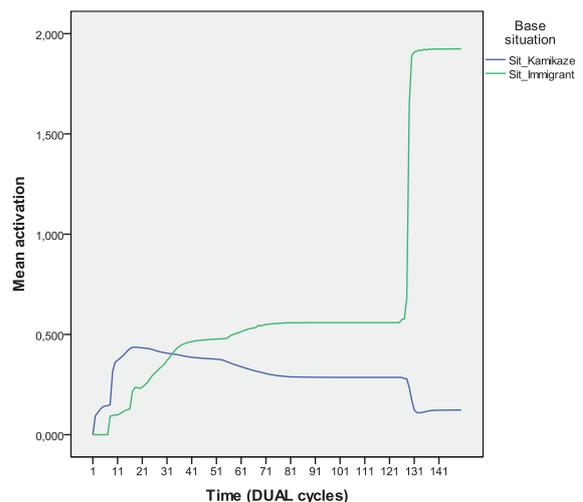


Figure 5. Activation level of the binding nodes for the two base situations (‘kamikaze’ and ‘immigrant’ as a function of time). At time 128 the mapping with the ‘Immigrant’ situation wins.

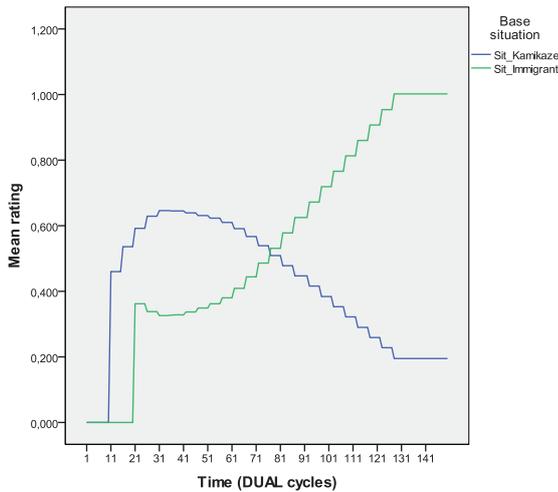


Figure 6. Rating of the hypotheses that the target situation ('terrorist') corresponds to the two base situations respectively ('kamikaze' and 'immigrant' as a function of time). At time 128 the mapping with the 'Immigrant' situation wins.

The transfer mechanism, however, does not wait for any winners. Soon after the goal-agent 'more acceptable act' from the 'terrorist' situation (see fig. 1) finds its correspondence, the system starts to transfer the respective relation. It is known from the base situation that the Bulgarian immigrant opened a Bulgarian restaurant in Ireland (which is an instance of popularization of the Bulgarian traditional culture) and this act causes stopping him beating of his wife. Thus, the most important causal relation (popularizing own culture in foreign countries causes acceptable actions) is transferred to the target situation (Figure 7). After winning of the respective analogy, these transferred agents remain in the description of the target and can be further interpreted.

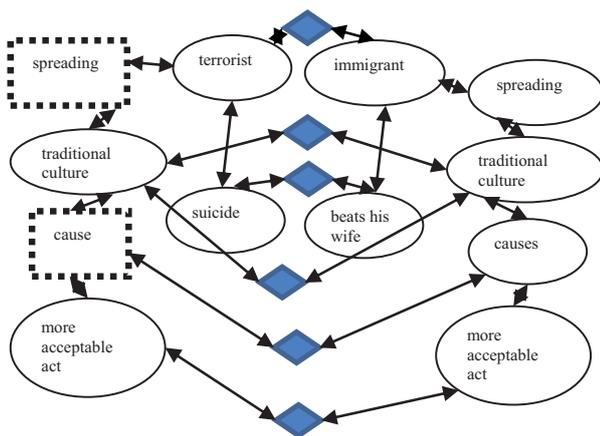


Figure 7. According to the transfer mechanism, if all arguments of a certain relation are mapped but the relation itself is not, then a copy of the respective relation is created. (The transferred elements are represented as dashed rectangles)

Experiment

The experiment is designed to test the model's prediction that a losing base for analogy may play role in highlighting specific aspects of the target that will improve the mapping between the target and another, appropriate base.

Design:

We performed an one-factorial between-group experiment. The independent variable was the group with two levels: control and experimental. The dependent variables were the judgments of the people on 7-point scales to four questions about how similar the stories and some of their aspects are.

Procedure:

Each participant received a sheet of paper with three short stories written on them. The instruction to the people was to read carefully all three stories and to prepare for answering some questions on them. There were no time limits for reading. Everybody worked alone, with the presence of the experimenter in the room only.

People from the control group received the stories "Terrorist", "Tsunami", and "Emigrant" (in this order); whereas people from the experimental group received "Terrorist", "Kamikaze", and "Emigrant" (see more about the stories in the section Stimuli below).

After that, the participants from the both groups received another sheet of paper with eight statements on each. The instruction was to evaluate on a 7-point scale how confident they feel each of the statements. The last four statements were equal for the both groups and concern the similarity between the "Terrorist" and "Emigrant" stories, as well the similarity between some of their aspects. The first four statements differed for both groups and concerned the similarity between the "Terrorist" and, respectively, "Tsunami" or "Kamikaze" stories. The subjects of analysis were the answers of the people to the four equal for both groups questions.

Stimuli

The four stories "Terrorist", "Kamikaze", "Tsunami", and "Emigrant" consisted of 120-170 words each. The first three stories were described as journalistic coverage, the fourth one – as a letter to a friend. The "Terrorist" coverage was about a lonely man who had crashed with a car-bomb in a market in New Jersey. The "kamikaze" report was about the grandson of a kamikaze, hero from the war. The grandson has been just nominated as an ambassador of Japan in US. The story for the tsunami (a control story for the participants from the control group only) was about a japan farmer who had lost his business because of a tsunami. The "Immigrant" story was a letter from the wife of the immigrant to her friend.

The questionnaire consisted of eight statements. The first four statements differed between the two groups. For the control group they served evaluating the similarity between the "Terrorist" and "Tsunami" stories; for the experimental group – respectively between the "Terrorist" and "Kamikaze" stories. People should evaluate how similar they feel the stories as a whole; the actions of the main heroes; the motives for their actions, and the nature of the persons as a whole.

The second group of four questions served for evaluating the similarity between the “Terrorist” and “Immigrant” stories according to the same criteria. These four questions were the same for both groups and were an object of analysis.

Participants:

42 students from New Bulgarian University participated in the experiment for course credits. They were randomly assigned to both groups. 24 of them fell in the control group; the other 18 – in the experimental group.

Results:

The main rating for how similar the stories for the terrorist and the kamikaze was 2.25 (st. dev. 1.225) for the control group, and 3.83 (st. dev. 1.79) for the experimental group. The difference turned to be significant: $t(40) = -3.404$, $p = 0.002$.

The respective differences for the three aspects of the stories, (whether the actions of the characters are similar; whether the motives for the actions of the characters are similar; whether the characters are similar in their nature) were not significant: respectively, $t(40) = -1.184$, $p = 0.243$; $t(40) = -0.798$, $p = 0.430$; $t(40) = -1.033$, $p = 0.308$.

Thus, the difference of the ratings for the overall similarity cannot be caused just by a simple assimilation effect. Instead, looking to each aspect of the stories separately, people from both groups do not differ in their ratings. However, it seems that people from both groups weight the different aspects of the stories differently because of the context of the third story. With other words, people weight the different aspects of the mapped stories differently because of the context. This means that they have different representations of the target situation.

Conclusions

Analogy-making is a powerful human ability for decision-making and evaluation. However, retrieval of the most appropriate base for analogy is a very difficult task both for humans and for the most of the models for analogy-making. It is relatively easy to retrieve situations that share the same superficial properties with the target, but it is very hard to retrieve a situation that shares the same high-level relations. In addition, the problem becomes even more difficult if the most important for the appropriate mapping aspects of the target story are not vivid.

We proposed an idea how both problems may be attacked via exploring the dynamics of multiple analogies. Instead of trying to retrieve the appropriate base directly, one may use one or more intermediate superficial analogies that slowly converge the system to the right solution. From one side, the intermediate analogies may help for the retrieval of a better structurally but less superficially similar episodes. From the other side, the intermediate analogies may cause a re-representation of the target and may highlight different aspects of it.

We used the AMBR model for analogy making to simulate this idea. One aspect of the representation of the target situation was left inactivated. The system easily

extracts from its memory one superficially similar base and launched the mapping process. It was impossible for it at the beginning to activate one more appropriate base for the analogy because of its remoteness.

However, we propose at least two ways of how this remote base may be activated indirectly:

First, the initial mapping with the superficial base may cause a re-representation of the target, highlighting the non-vivid aspects of it.

Second, the superficial base may help for the further spreading of the activation to close and far associations.

The mechanisms for structural correspondence of the MABR model allow it to support and maintain the structurally well-organized mappings. Thus, nevertheless that the activation may spread to very different basis and many different initial mappings may be launched, AMBR behaves stable enough. Once it finds the most appropriate base, the consistent mappings cause additional activation of the respective appropriate base.

The hypothesis that a third, structurally not good base, may facilitate the analogy between two situations was tested with a psychological experiment. People judged with higher ratings the similarity between two situations in the context of a carefully chosen third one, in comparison with the same judgments in the context of an arbitrary third story. The context was chosen in a way to initiate some mappings between the target and the contextual stories. These initial mappings should make the important aspects of the target story on which the two stories differ more vivid. As a consequence, people weight these aspects higher.

At the same time, if people focus on the similarity of a certain aspect of the stories, there is no reason the context to influence their ratings. This was confirmed by the experimental results – people’s ratings differ depending on the context only when the similarity of the whole stories should be evaluated; not when the respective similarity between concrete aspects of the stories should be rated.

Acknowledgements

This research was supported financially by the European Office for Aerospace Research and Development under grant FA8655-10-1-3061 (Adaptive Problem Solving by Analogy).

References

- Burstein, M. (1988). Incremental Learning from Multiple Analogies. In: Prieditis, A. (ed.) *ANALOGICA*, Pitman, London.
- Burstein, M. (1988). Combining Analogies in Mental Models. In: D. Helman (ed.), *Analogical Reasoning*, Kluwer Academic Publishers, pp. 179-203.
- Burstein, M. (1986). Concept Formation by Incremental Analogical Reasoning and Debugging. In: *Machine Learning: An Artificial Intelligence Approach*, Volume II, Michalski, R. S., Carbonell, J. G., and Mitchell, T. M., eds., Morgan Kaufmann Publishers, Inc., Los Altos, Ca., pp. 351-370.

- Burstein, M. (1986). Analogical Learning with Multiple Models. In: Mitchell, T. & Carbonell, J. (eds.) *Machine learning: a guide to current research*. Kluwer,
- Clement, J. (1993). Using Bridging Analogies and Anchoring Intuitions to Deal with Students' Preconceptions in Physics. *Journal of Research on Science Teaching*, vol. 30, pp. 1241-1257
- Clement, J. (2009). *Creative Model Construction in Scientists and Students: The Role of Imagery, Analogy, and Mental Simulation*. Springer
- Collins, A., Gentner, D. (1987). How people construct mental models. In D. Holland & N. Quinn (Eds.), *Cultural models in language and thought* (pp. 243-265). England: Cambridge University Press.
- Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7, 155-170.
- Gentner, D. (1989). Mechanisms of analogical learning. In S. Vosniadou and A. Ortony, (Eds.), *Similarity and Analogical Reasoning*, 199-241. London: Cambridge University Press.
- Gentner, D., Holyoak, K., Kokinov, B., eds. (2001). *The Analogical Mind: Perspectives from Cognitive Science*. Cambridge, MA: MIT Press.
- Gentner, D., Markman, A. (1997). Structure Mapping in Analogy and Similarity. In: *American Psychologist*, vol. 52, pp. 45-56
- Gentner, D. et al. (1997). Analogical Reasoning and Conceptual Change: A Case Study of Johannes Kepler. *The Journal of Learning science*, vol. 4, pp. 3-40.
- Falkenhainer B., Forbus K., & Gentner D. (1989). The structure mapping engine: Algorithm and examples. *Artificial Intelligence*, 41 (1), 1-63
- Hofstadter, D. & the Fluid Analogies Research Group (1995). *Fluid concepts and creative analogies*. New York: Basic Books
- Hofstadter, D. (2001). Analogy as the Core of Cognition. In: Gentner, D., Holyoak, K., Kokinov, B. (eds.) *The Analogical Mind: Perspectives from Cognitive Science*. Cambridge, MA: MIT Press
- Holyoak K. & Thagard P. (1989). Analogical mapping by constraint satisfaction. *Cognitive Science*, 13, 295-355
- Holyoak, K., Gentner, D., Kokinov, B., eds., (1998). *Advances in Analogy Research: Integration of Theory and Data from the Cognitive, Computational, and Neural Sciences*. Sofia: NBU Press
- Holyoak, K., Gentner, D., Kokinov, B. (2001). The Place of Analogy in Cognition. In: Holyoak, K., Gentner, D., Kokinov, B. (eds.) *The Analogical Mind: Perspectives from Cognitive Science*. Cambridge, MA: MIT Press.
- Hummel, J. & Holyoak, K. (1997). Distributed representation of structure: A theory of analogical access and mapping. *Psychological Review*, 104, 427-466.
- Kokinov, B. (1994). A Hybrid Model of Reasoning by Analogy. Chapter 5. in: K. Holyoak & J. Barnden (eds.) *Analogical Connections, Advances in Connectionist and Neural Computation Theory*, vol.2, Ablex Publ. Corp.
- Kokinov, B. Petrov, A. (2000). Dynamic Extension of Episode Representation in Analogy-Making in AMBR. In: *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*. Erlbaum, Hillsdale, NJ.
- Kokinov, B., Petrov, A. (2001). Integration of Memory and Reasoning in Analogy-Making: The AMBR Model. In: Gentner, D., Holyoak, K., Kokinov, B. (eds.) *The Analogical Mind: Perspectives from Cognitive Science*, Cambridge, MA: MIT Press.
- Kokinov, B., Vankov, I., Bliznashki, S. (2009). How Analogy Could Force Re-representation of the Target and Inhibition of the Alternative Interpretation. In: Kokinov, B., Holyoak, K., Gentner, D. (eds.). *New Frontiers in Analogy Research*. Sofia: NBU Press
- Kokinov, B., Bliznashki, S., Kosev, S., Hristova, P. (2007). Analogical Mapping and Perception: Can Mapping Cause a Re-Representation of the Target Stimulus? In: *Proceedings of the 29th Annual Conference of the Cognitive Science Society*. Erlbaum, Hillsdale, NJ.
- Kokinov, B., Holyoak, K., Gentner, D. eds. (2009). *New Frontiers in Analogy Research*. Sofia: NBU Press
- Shelley, C. (1998). Multiple analogies in evolutionary biology. In: Holyoak, K., Gentner, D., Kokinov, B. (eds.) *Advances in Analogy Research: Integration of Theory and Data from the Cognitive, Computational, and Neural Sciences*. Sofia: NBU Press.
- Shelley, C. (1999). Multiple Analogies in Archeology. *Philosophy of Science*, vol. 66, pp. 579-605
- Shelley, C. (1999). Reasoning by Multiple Analogies. PhD dissertation. U. Waterloo.
- Shelley, C. (2003). *Multiple Analogies in Science and Philosophy*. John Benjamins Publishing
- Spiro, R., Feltovich, P., Coulson, R., and Anderson, D. (1989). Multiple analogies for complex concepts: antidotes for analogy-induced misconception in advanced knowledge acquisition. In: Vosniadou & Ortony (eds.) *Similarity and Analogical Reasoning*. Cambridge Univ. Press, pp. 498-531
- Vamvakoussi, X. & Vosniadou, S. (in press). Bridging the gap between the dense and the discrete: The number line and the 'rubber line' bridging analogy. *Mathematical Thinking and Learning*.
- Veloso, M.M. & Carbonell, J.G.(1993). Derivational Analogy in PRODIGY: Automating Case Acquisition, Storage, and Utilization. *Machine Learning*, 10, 249-278
- Vosniadou, S., & Skopeliti, I. (in press). Instructional Analogies in Conceptual Restructuring Processes. *Cognitive Psychology*.

A Cognitive Model For Predicting Aesthetical Judgements As Similarity to Dynamic Prototypes¹

Ute Schmid (ute.schmid@uni-bamberg.de)

Michael Siebers (michael.siebers@uni-bamberg.de)

Johannes Folger (johannes.folger@stud.uni-bamberg.de)

Simone Schineller (simone.schineller@stud.uni-bamberg.de)

Dominik Seuß (dominik.seuss@stud.uni-bamberg.de)

Cognitive Systems, Faculty Information Systems and Applied Computer Science
University of Bamberg, Germany

Marius Raab (marius.raab@uni-bamberg.de)

Claus-Christian Carbon (ccc@uni-bamberg.de)

Stella J. Faerber (stella.faerber@uni-bamberg.de)

Department of General Psychology and Methodology
University of Bamberg, Germany

Abstract

We present a framework for cognitive modeling of aesthetic decision making based on dynamic prototypes. Starting point of our work is empirical evidence which shows that subjects' initial ratings of attractiveness of objects can be influenced by adapting them to new, typically more innovative objects. The framework consists of three steps: (1) Estimating an initial prototype from the ratings, (2) adapting the prototype due to the impact of the new objects, and (3) predicting the attractiveness ratings for subsequently presented object by their similarity to the adapted prototype. The framework allows representation of prototypes and objects as feature vectors containing metrical or categorical attributes or as structural representations. Within the framework, a variety of similarity measures and similarity-to-rating mappings can be explored to gain more precise insight in the cognitive processes underlying aesthetical appreciations. We instantiated the framework for a first set of data obtained in a psychological experiment. In this experiment subjects rated the attractiveness of an initial set of chairs which varied in length of the backrest and the saturation of the color. Subjects then were adapted to a new set of chairs with extreme values on both dimensions. Finally, subjects again rated the initial objects. We tested our model and obtained promising first results.

Introduction

Aesthetical judgements are not only underlying the evaluation of works of art but also guide our purchase decisions for mundane objects (Whitfield & Slatter, 1979;

Hekkert, Snelders, & Wieringen, 2003). Whenever we buy something – may it be clothing, furniture, a phone, or a car – our decision is influenced by aesthetical aspects. That is, given a class of objects with comparable functionality, price range, and brand image, we still prefer one object over another. Often, this preference is based on visual cues and, more often than not, we cannot give a clear justification for our preference.

One possible explanation for such aesthetical preferences is the similarity of objects to our individual prototype for the object category (Rosch, 1978; Kruschke, 2008). Such prototypes are constructed over personal experience and therefore dynamic (Medin & Heit, 1999; Ashby & Maddox, 2005). This is reflected, for example, in the way we are affected by changes of fashion. The majority of people typically does not like a new style in clothing or car design if it is freshly introduced to the market. However, if they are exposed to the new design over some time, their aesthetical judgement adapts and the previously liked designs appear less attractive while the new design gains attractiveness (Carbon, 2010).

Experimental evidence for adaptation effects in aesthetical judgements was, for example, given by Faerber and Carbon (2010). An experimental procedure for an adaptation experiment can be realized in the following way: Initially (T_1), subjects are presented a set of stimuli (e.g., chairs) which vary on some dimensions (e.g., length of backrest and saturation of color, see Fig. 3). Some objects are similar to standard – that is, prototypical – artefacts, others highly deviate from typical appearance. Subjects have to rate the attractiveness of the given objects. In a second phase (adaptation phase

¹The reported results were obtained in a student project of J. Folger, S. Schineller, and D. Seuß, supervised by Ute Schmid and Michael Siebers. The collaboration between the Cognitive Systems group of Ute Schmid and the General Psychology Group of Claus-Christian Carbon is supported by a grant of *Bayerisches Staatsministerium für Wissenschaft, Forschung und Kunst*.

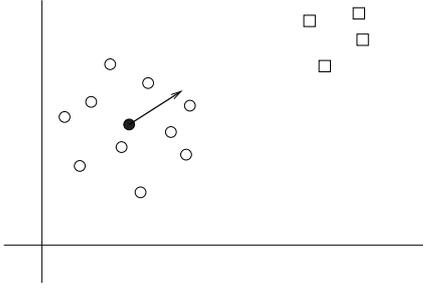


Figure 1: Illustration of a prototype-shift in feature space due to adaptation to novel objects (represented as squares)

A), subjects are induced to engage with artefacts which deviate not, moderately or strongly from the typical objects. For example, they have to rate different functional and aesthetical features of these objects. Afterwards (T_2), subjects have to rate the attractiveness of the objects in the initial set again. Over several experiments, Carbon and his coworkers could show, that if subjects were engaged with strongly deviating objects during the adaptation phase, at T_2 the more deviating stimuli are rated more attractive as at T_1 while the more standard objects are rated less attractive.

Carbon and colleagues explain this effect by recalibration or dynamic prototype change (see Fig. 1): When confronted with a new artefact which deviates too much from the prototype for this class of objects (e.g., very angular car shape, belly-bottom trouser legs), such new artefacts are rated as not attractive (T_1). However, if one gains more experience with such innovative objects (A), the prototype undergoes a dynamic change, incorporating the new objects. Consequently, after a while (T_2), the objects which were originally similar to the prototype at (T_1) are now more distant and the objects which originally strongly deviated from prototype are now similar to the updated prototype (T_2).

To gain more precise insights in the dynamic changes of prototype representations and their impact on aesthetical decision making, we propose a cognitive modeling framework which allows (1) to estimate an initial prototype from aesthetical judgements of objects at the time of the first exposure (T_1), (2) to adapt this initial prototype with respect to the adaptation set (A), and (3) to use this prototype to predict subsequent aesthetical judgements of objects (T_2). Such a model can help to gain a deeper understanding of aesthetical decision making. Furthermore, it can provide an initial building block for an assistant system which allows designers to evaluate the possible market success of new design lines.

In the following, we first propose a general framework for prototype based generation of aesthetical judgements. Afterwards, we present a first instantiation of the

framework where we model data gained from a psychological experiment. We conclude with a short discussion and further work to be done.

A Framework for Generating Aesthetical Judgements

Given the proposition that an individual generates his/her aesthetical judgement of an object with respect to its similarity to his/her prototype, the general framework can be expressed as

$$\forall o \in O : K(\sigma(o, p)) = a(o) \quad (1)$$

where $\sigma(o, p)$ is the similarity of the object o to prototype p , K is a kernel function, and $a(o)$ is the resulting attractiveness rating for the object. To simplify matters, we do not discriminate between $a(o)$ as the mental representation of the attractiveness of the object and $a(o)$ as the externally expressed judgement which, for instance, is given as a rating on a Likert scale.

To instantiate the general approach, the following questions must be answered:

- What kind of information of the real-world objects is included in the prototype?
- How is the prototype represented?
- With what type of measure is the similarity between prototype and object established?
- Which kernel function is used to map the similarity to the attractiveness rating?

Illustration

We illustrate these aspects using the material which will be presented in more detail in section Experiment. The objects under consideration are chairs. A chair might be represented using

- **holistic visual information** such as shape, which characterize a chair as elegant, comfortable, etc.
- **metrical visual features** such as length of the backrest,
- **metrical visual relations** such as the proportion of length of the backrest to depth of the seat,
- **metrical non-visual features** such as weight,
- **categorical visual features** such as color (which typically is perceived qualitative and not as a metrical feature representing wave length),
- **categorical non-visual features** such as producing country,
- **qualitative spatial relations** such as that the back legs of the chair are *under* the backrest or *in front* of the backrest.

Each subset of these different types of information implies a different representational format (Schmid et al., 2011). If only metric features are considered, each object can be represented as a feature vector and the prototype can be represented by an average value for each feature.

Under the – in most domains not valid – assumption (Nosofsky, 1988), that the features are not correlated and that the variability of feature values is comparable, a standard distance metric, such as Euclidian distance or Manhattan distance could be used to calculate the similarity between an object and a prototype. However, it is an open question, whether one of these measures is guiding the mental similarity assessment or whether more complex similarity measures are needed. Maybe, different features have different salience which would result in a measure with different weights for the different features. In general, the similarity measure should not only take into account the isolated features but also interaction terms.

Finally, there are many possible mappings from similarity to aesthetical judgements. In the most simple case, this might be a linear regression $\beta_0 + \beta_1(\sigma(o, p)) = a(o)$. In the case of a similarity measure which deals with different components of object-representation differently, σ and β_1 might be vectors. Alternatively, the mapping might be non-linear and only captured by specific non-linear functions. A typical observation is that ratings of attractiveness are based on the MAYA (most advanced yet acceptable) principle (Hekkert et al., 2003). That is, objects which are very similar to the prototype are not perceived as highly, but only medium attractive (because they are somewhat boring) and objects which deviate too far from the prototype are considered as highly unattractive.

The proposed general model can be viewed as a guideline for exploring empirical data to obtain more specific information about the processes underlying aesthetical decision making.

Identifying the Similarity and Mapping Functions

In the context of an experimental setting researching adaptation as described above (see sect. Introduction), the ratings obtained during initial representation (T_1) of objects are used to determine $K(\sigma(o, p))$ in such a way that the ratings of each individual can be reproduced as exactly as possible. To identify σ and K , we propose the following procedure:

- Predefine a set of plausible measures $\Sigma = \{\sigma_1, \dots, \sigma_n\}$ and functions $\kappa = \{K_1, \dots, K_m\}$.
- For each combination $K_j(\sigma_i(o, p))$ estimate p such that the prediction error of $a(o)$ is minimal over all objects o in O_1 . How the estimation can be performed depends on the form of σ_i and K_j . In the most simple case, it might be possible to gain the estimate analytically. Alternatively, the prototype values could be identified by gradient descent, or –

if non-derivable functions are involved – by Monte Carlo studies.

- Select the most simple function K_j and measure σ_i which produces minimal errors.

We believe it reasonable to assume that the functions found to be fitting the individual ratings best should be kept constant for the attractiveness ratings after the prototype adaptation phase (at T_2).

Predict Aesthetical Judgements Due to Dynamic Shift of the Initial Prototype

To include the dynamic change of the initial prototype due to adaptation to novel objects, the framework is extended to

$$\forall o \in O : K(\sigma(o, S(p, O_A))) = a(o) \quad (2)$$

where $S(p, O_A)$ is a function modeling the shift of the initial prototype due to adaptation.

The form of the shift function is dependent on the similarity measure and mapping function obtained from the initial ratings. If, for instance, the similarity measure is based on independent, equally salient features and the kernel is a linear function, than the prototype is shifted in the direction of the feature vector of the average over all objects in the adaptation phase (see Fig. 1). However – again – things can get more complex. Therefore, different shift functions S should be investigated in the context K_j and σ_i identified in the previous step. The general procedure for selecting a suitable S is analogous to the previous step.

Experiment

The stimuli used in the experiment are chairs which were constructed by varying length of the backrest ($l(o)$) and saturation ($s(o)$). A matrix of chairs where length and saturation is varied in ten equi-distant steps is given in Figure 2. For the experiment, chairs for every second variation were selected as test sets – that is, saturations are -60, -30, 0, 30, 60 and lengths are 1, 3, 5, 7, 9. This selection was made to ensure that the visual variations were perceivable when presenting the objects at a computer monitor. To refer to a specific chair o , we give its feature vector $\langle l(o), s(o) \rangle$.

21 subjects participated in the study. In a first session (T_1), subjects rated each of the 25 chairs of the test set on a 7-step Likert scale. Afterwards (A), subjects were adapted to four chairs with extreme values: the most extreme chair with $\langle 9, -60 \rangle$ was already contained in the test set, the other three chairs were the neighbours $\langle 8, -60 \rangle$, $\langle 8, -45 \rangle$, and $\langle 9, -45 \rangle$ (see Fig. 3). After a time-lag of seven days, this adaptation set was presented again (A) and afterwards (T_2), attractiveness ratings for the 25 test chairs were obtained the second time.

The experiment was not specifically designed to explore our cognitive framework. With one rating for each

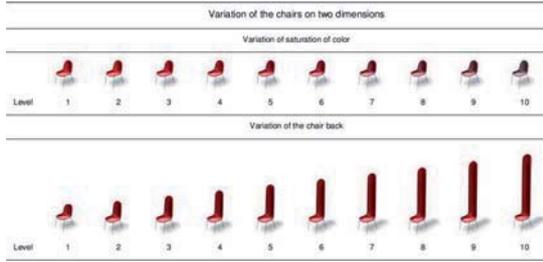


Figure 2: Variations of length and saturation

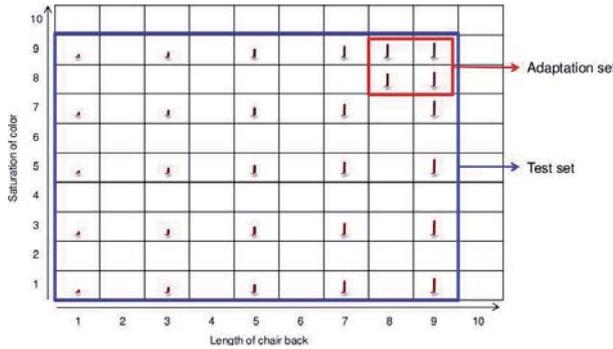


Figure 3: Object space with 10 equi-distant variations of the two dimensions length of chairback and saturation (only every second variation per dimension was included in the initial object set)

of the 25 chairs in the test set, we have a rather small number of data available for individual models. Consequently, we can only explore similarity measures and mapping functions which involve a small number of free parameters. Furthermore, it can be assumed that saturation is perceived more dominantly for chairs with longer backs than for chairs with shorter ones. Finally, there might be some impact of the amount of space taken by a presented chair in relation to the background. With these caveats, we now will present the cognitive models.

A Model For Attractiveness Ratings of Chairs

To generate a model based on our framework presented in equation 1, the values of length and saturation were normalized via z-transformation.

Excluding a Simple Linear Model

Applying the Occam's razor principle of simplicity, the first choice for modeling was to assume that ratings of attractiveness are linearly dependent on similarity. That there is no simple linear relation between prototypicality and attractiveness is obvious from the interaction diagrams of $l(o)$, $s(o)$ and $a(o)$ (see Fig. 4).

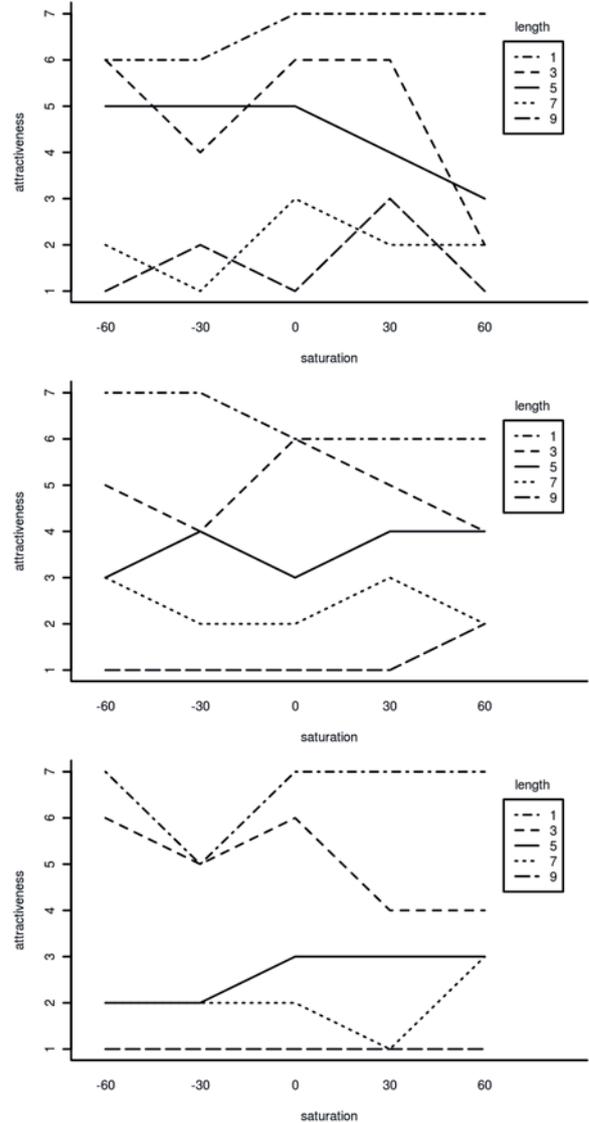


Figure 4: Interaction between length $l(o)$, saturation $s(o)$ and initial attractiveness rating $a(o)$ for three subjects

Approximating a First Model for the Initial Prototype

To capture the non-linear effect of variations in length and saturation, we propose the following instantiation of the framework given in equation 1:

$$\sigma(o, p) = \begin{pmatrix} |l(o) - l_p| \\ |s(o) - s_p| \end{pmatrix} \quad (3)$$

$$K(\langle x_l, x_s \rangle) = \beta_0 + \beta_1 e^{-x_l} + \beta_2 e^{-x_s} + \beta_3 e^{-x_l x_s} \quad (4)$$

Using the e function (instead of a polynomial) is reasonable because it results in the fewest possible number of free parameters in the models. When taking into account two feature dimensions and their interaction the minimal number of free parameters is 4. The initial prototypes were estimated by minimizing

$$\frac{1}{2} \sum_{i=1}^{25} (a(o_i) - K(\sigma(o_i, p)))^2 \quad (5)$$

for each subject where o_i and p are vectors $\langle l, s \rangle$. The values for the initial prototypes $\langle l_p, s_p \rangle$ were calculated using gradient descent with decaying learning rate η with initial value $\eta = 0.025$ and momentum $\alpha = 0.25$ iterating over 500 cycles. Values l_p and s_p were initialized to the means of the highest rated objects.

The estimated prototypes produced acceptable small deviations between predicted and observed attractiveness ratings (see Tab. 1). The estimated initial prototypes are given in Figure 5. Note, that there are three subjects (11, 15, 21) who preferred chairs with long back-rests from the beginning.

Predicting Attractiveness Ratings From the Shifted Initial Prototypes

Given the estimates for the individual initial prototypes, in the next step the model was applied to predict the attractiveness ratings after shifting the initial prototype due to the adaptation set. Equation 2 as proposed above was used for estimation. The parameters β estimated for the initial prototype were kept as it is reasonable to assume that the individual influence of the different features is constant within subjects. Again, gradient descent was applied with initial $\eta = 0.0005$.

With the exception of three subjects (6, 11, 15), the predicted attractiveness ratings again have acceptable small deviations from the observed ratings. For these three subjects it might be possible that the good fit for the initial prototype was due to a local minimum.

The prototype shifts are given in Figure 5. For the majority of subjects the shift is in the direction of longer chairs. This is plausible because the adaptation set consisted of four chairs with lengths 8 and 9. Only subjects 11, 17, and 21 show a shift towards shorter lengths. However, this shift is very small for 17 and 21. In the direction of saturation (which was -60 and -45 in the

Table 1: Estimated values $\langle l_p, s_p \rangle$ for the initial prototype and estimated values $\langle l_s, s_s \rangle$ for the shifted prototype with mean squared residuals

Pb	l_p	s_p	MSSQ(a_1)	l_s	s_s	MSSQ(a_2)
1	1.42	41.08	17.29	1.75	41.07	39.22
2	1.00	-0.07	6.81	1.73	-0.004	48.51
3	1.54	9.67	27.22	2.79	-0.005	65.62
4	1.43	32.34	37.58	1.56	32.34	24.10
5	1.56	-51.09	10.19	2.13	-51.09	17.67
6	1.66	59.58	13.97	3.91	59.61	100.42
7	2.24	57.63	10.57	2.37	57.65	25.88
8	1.95	51.28	7.64	2.53	32.35	45.06
9	1.25	-4.78	29.46	3.00	-5.18	31.66
10	1.99	-50.52	18.93	5.00	-50.51	45.09
11	6.38	48.60	9.91	5.00	30.00	108.15
12	1.54	9.78	11.79	1.60	18.31	23.93
13	3.88	-35.14	17.86	4.74	-35.13	34.03
14	1.53	55.55	11.26	1.89	55.55	89.45
15	8.21	-56.70	44.86	8.33	-56.75	103.34
16	1.97	-59.97	14.97	3.14	-58.13	24.49
17	1.33	-25.16	2.66	1.23	-25.16	11.77
18	3.72	54.17	8.34	4.02	47.23	14.09
19	2.17	40.46	38.44	2.25	40.46	37.94
20	1.00	-44.86	6.02	1.90	-44.86	33.56
21	7.05	29.91	19.24	6.72	29.00	21.11

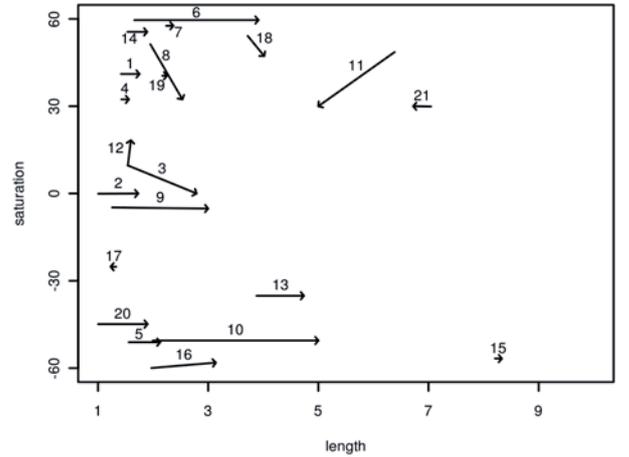


Figure 5: Initial prototypes and their shift which predict initial attractiveness ratings and attractiveness ratings after adaptation

adaptation set) there is no clear pattern for the shift. This might be due to the fact that the visual salience of saturation is more variable between subjects than the visual salience of length.

Conclusion

Given empirical findings which demonstrate that aesthetic preferences change dynamically over time, we proposed a cognitive framework. Within this framework it is claimed that aesthetic judgements are based on similarity to prototype. Similarity assessment and mapping of similarity to attractiveness are proposed as sub-processes underlying aesthetic decision making. The framework therefore gives a guideline to explore empirically which types of similarity measures and mapping functions are realised when subjects perform ratings of

attractiveness.

We explored the framework with empirical data which were obtained in an experiment where subjects rated the attractiveness of chairs which varied in the length of the backrest and the saturation of color. Although there were only 25 data points per subject, we got satisfying results in predicting the shift of aesthetical judgements due to adaptation to novel stimuli.

Based on this initial work, there are several aspects which we plan to explore in future work: In the current model the shift of the prototype is estimated in a single time step over all objects of the adaptation set. A psychological more plausible approach would be to model an *incremental* shift. However, for an incremental model, it is necessary to determine in advance (a) the degree of the shift – that is, how strong a new object pulls the prototype in its direction – and (b) the direction of the shift – that is, the possible different weights of the dimensions in the object space. Such an incremental model would have an additional advantage since it allows a new way to combine empirical evidence of mere exposure respectively the exemplar theory of categorization and prototype theory: Because each presented object induces a shift, the prototype updates are sensitive not only to variations in object attributes but also to frequency of object presentation. That is, if the same objects is presented several times, each presentation would induce a shift.

Another aspect we plan to explore in the future is to investigate more sophisticated measures of similarity, e.g., using different similarity measures for the different aspects of the objects. Another alternative could be to replace the similarity measure by fuzzy memberships. Furthermore, we are interested in models which capture a mixture of metrical and categorial features and in models which capture the holistic visual impression.

Finally, the experiment was not specifically designed to test the proposed framework. Therefore, we plan to conduct more specific experiments to explore the explanatory power of our framework. Especially, we plan to investigate attractiveness ratings when object appearance is varied on different kinds and numbers of dimensions. Stimuli should be obtained from different artificial and natural domains.

References

- Ashby, F., & Maddox, W. (2005). Human category learning. *Annual Review of Psychology*, *56*, 149-178.
- Carbon, C. C. (2010). The cycle of preference: long-term dynamics of design properties. *Acta Psychologica*, *134*(2), 233-244.
- Faerber, S. J., & Carbon, C. C. (2010). On the role of visual adaptations on aesthetic appreciation via changes of the object space. *Perception*, *39*(S), 76.
- Hekkert, P., Snelders, D., & Wieringen, P. van. (2003). ‘Most advanced, yet acceptable’: Typicality and novelty as joint predictors of aesthetic preference in industrial design. *British Journal of Psychology*, *94*, 111-124.
- Kruschke, J. K. (2008). Models of categorization. In R. Sun (Ed.), *The cambridge handbook of computational psychology* (p. 267-301). New York: Cambridge University Press.
- Medin, D., & Heit, E. (1999). Categorization. In D. Rumelhart & B. Martin (Eds.), *Handbook of cognition and perception* (p. 99-143). San Diego: Academic Press.
- Nosofsky, R. M. (1988). Similarity, frequency, and category representations. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *14*(1), 54-65.
- Rosch, E. (1978). Principles of categorisation. In E. Rosch & B. Lloyd (Eds.), *Cognition and categorization* (p. 2748). Hillsdale NJ: Lawrence Erlbaum.
- Schmid, U., Grossmann, P., Wachter, M., Raab, M., Carbon, C.-C., & Faerber, S. J. (2011). How visible are different variations of spatial features and relations in logos and how does visibility affect prototype generation? In *Post-Proceedings of the KI'11 Workshop Visibility in Information Spaces and in Geographic Environments (Berlin, Oct. 4, 2011)* (p. 25-36). University of Bamberg Press.
- Whitfield, T., & Slatter, P. (1979). The effects of categorization and prototypicality on aesthetic choice in a furniture selection task. *British Journal of Psychology*, *70*, 65-75.

An ACT-R Approach to Reasoning about Spatial Relations with Preferred and Alternative Mental Models

Sven Brüßow (sven@cognition.uni-freiburg.de)¹

Matthias Frorath (frorath@informatik.uni-freiburg.de)¹

Marco Ragni (marco.ragni@cognition.uni-freiburg.de)¹

Thomas Fangmeier (thomas.fangmeier@uniklinik-freiburg.de)^{1, 2}

¹Center for Cognitive Science, University of Freiburg

²Department for Psychiatry, University Clinic Freiburg

Abstract

A computational model of spatial relational reasoning implemented in the ACT-R cognitive architecture allows for the simulation of a wide range of behavioral data in the context of both determinate and indeterminate deductive spatial reasoning tasks. In that respect the presented study bridges the gap between results of previous work that investigated determinacy conditions separately. ACT-R's subsymbolic processing principles substantially contribute to the underlying theory of Preferred Mental Models as they add a powerful component making precise accuracy predictions possible. In addition, the data is informative about a possible strategy when the task is to judge if an externally presented spatial description matches a mental model that resulted from the current reasoning process.

Keywords: Spatial reasoning; mental models; ACT-R

Introduction

The objective of psychological theories to explain and predict experimental data is best realized by cognitive models implemented in cognitive architectures like ACT-R (Anderson et al., 2004; Anderson, 2007). Cognitive models help researchers develop intuitions about the cognitive demands of certain tasks, generate additional data that can be tested against human data, and eventually allow for theory reevaluation. ACT-R is empirically grounded. Its explanatory power lies in the combination of discrete symbolic descriptions with constantly interfering non-discrete subsymbolic processes. Both concepts are needed for an implementation of a psychologically plausible theoretical account.

In the present work, we investigated the potential of ACT-R to substantiate the theoretical framework of the Preferred Mental Model Theory (PMMT). The PMMT describes the deduction process in the context of ambiguous descriptions and stands in the tradition of the classical Mental Model Theory originally introduced by Johnson-Laird (1980, 1983). Mental models are constructed from given spatial relational information that is typically given by a set of premises. The PMMT suggests distinct construction principles for determinate and indeterminate premises. If premises are determinate they allow for only one model (which we call UNI for *unique*). If premises are indeterminate they allow for multiple model derivations. In this case some mental models derived from these premises may be considered in the deduction process while others are neglected. The mental model that is preferred over alternatives will be referred to as the preferred mental model (PMM), the first alternative model is denoted by AM1, and the second by AM2.

The reasoning process is commonly divided into three distinct phases. First, in the *construction phase* the initial PMM is incrementally built up from the given premises. Second, during the *inspection phase* evidence shows that human reasoners try to use the spatial information encoded in the PMM to validate a putative conclusion (Knauff, Rauh, & Schlieder, 1995; Rauh, Hagen, Schlieder, Strube, & Knauff, 2000; Rauh et al., 2005; Jahn, Knauff, & Johnson-Laird, 2007); in accordance with the PMMT the PMM involves the lowest construction costs (Ragni, Knauff, & Nebel, 2005). Third, in the *variation phase*, a correct inference depends on the type of mental model taken into account; dependence solely upon the initial PMM can lead to counter-examples being missed. The respective inference process is formally described by discrete operations starting with incremental integration of premise terms into the PMM; followed by model-conclusion comparisons; and, if necessary, continued with PMM modifications to an alternative mental model (Ragni & Brüßow, 2011). Alternative models can be generated by applying additional variation processes. Hence, with respect to the indeterminate premises illustrated in Table 1, to validate the conclusion “is D to the left of B?” two transformations are necessary, provided that participants test the conclusion with the possible mental models in the predicted order of PMM, followed by AM1, and then AM2.

Although the model variation phase is important, little research has systematically investigated the underlying processes. Rauh et al. (2000) were the first to investigate errors of omission and commission in spatial relational reasoning with intervals. Their presented theory, however, is purely symbolic; it does not allow for memory effects or other subsymbolic effects. This is why we decided to implement the PMMT in ACT-R.

ACT-R is both a theory and architecture of cognition. It has successfully been used to simulate a wide range of cognitive tasks. Standing in the tradition of production systems, originally introduced by Newell and Simon (1972) as appropriate formalisms for describing human problem solving behavior, procedural knowledge is described by sets of production rules that operate on memory chunks that in turn represent declarative knowledge. Modularly organized, ACT-R provides distinct components each specialized for certain perceptual or cognitive tasks; environmental visual information is processed in the vision-module, internal goal formulation

Table 1: Determinate premises resulting in the model “ABCD” (top) and indeterminate premises resulting in one of the three mental models “ABCD,” “ACBD,” or “ACDB” (bottom). Model denomination refers to unique model (UNI), preferred mental model (PMM) or alternative mental models (AM1, AM2). \overrightarrow{AB} represents the premise “A is to the left of B” and \overleftarrow{AB} the premise “B is to the right of A.” The remaining premises should be read accordingly. Arrows, in addition, indicate the order of term presentation. Double curve arrows above mental model representations mark those terms that need be transposed to transform the source model to the next alternative model.

P1	P2	P3	UNI	PMM	AM1	AM2
\overrightarrow{AB}	\overrightarrow{BC}	\overrightarrow{CD}				
\overleftarrow{AB}	\overleftarrow{BC}	\overleftarrow{CD}				
\overrightarrow{AB}	\overleftarrow{BC}	\overrightarrow{CD}	ABCD	\emptyset	\emptyset	\emptyset
\overleftarrow{AB}	\overleftarrow{BC}	\overleftarrow{CD}				
\overrightarrow{AB}	\overrightarrow{AC}	\overrightarrow{CD}				
\overleftarrow{AB}	\overrightarrow{AC}	\overrightarrow{CD}				
\overrightarrow{AB}	\overleftarrow{AC}	\overrightarrow{CD}	\emptyset	ACBD	ACBD	ACDB
\overleftarrow{AB}	\overleftarrow{AC}	\overrightarrow{CD}				

is constructed in the goal module, problem state information is processed by the imaginal module, the retrieval module accesses information stored in declarative memory, the manual module performs motor responses, and the procedural module represents the central executive controlling the activities of these modules. Each module is equipped with an interface called a buffer holding single chunks that are thus available across modules. In addition to the capacity restriction of just one chunk per buffer, productions process information in a strictly serial way; only one production may be active at a time. Note that this is not a general restriction of production systems. Modules, however, may be active in parallel; those chunks currently distributed across buffers are processed simultaneously.

A distinguishing feature of ACT-R is that the above described symbolic behavior is constantly directed by subsymbolic probabilistic processes. For example, the availability of chunks to the retrieval buffer depends on their level of activation, a dynamic numeric value that is computed for each chunk. Once a chunk has been created its initial activation decreases according to a fixed decay rate. Chunk activation, however, can also increase dependent upon (i) the number of positive retrievals in the past, (ii) spreading activation from other chunks in the buffers, and (iii) merging of identical chunks. Similarly, production selection depends on the utility value associated with each production; if multiple productions match simultaneously, their utility value controls ambiguity resolution.

Although the recent focus of research in the context of ACT-R as a theory has definitely been on grounding the architecture on a neurological basis, traditional wide range coverage of behavioral data predictions yet remains an inevitable prerequisite. Earlier versions of the presented model were reported previously (Ragni, Fangmeier, & Brüssow, 2010; Ragni & Brüssow, 2011). It was originally developed to test ACT-R’s BOLD function (Anderson, 2007) with fMRI data obtained from a study by Fangmeier, Knauff, Ruff, and Sloutsky (2006) and tested with more sophisticated behavioral data obtained from a study by Ragni, Fangmeier, Webber, and Knauff (2007). Fangmeier et al. investigated phases of the reasoning process using only determinate tasks whereas the study of Ragni et al. reported data restricted to indeterminate tasks. Furthermore, to inhibit linguistic processes Fangmeier et al. presented premises and conclusions as abstract terms whereas Ragni et al. presented complete sentences. Hence, it remained desirable to test model predictions for both uniformly presented determinate and indeterminate material obtained from one and the same study.

Accordingly, we present data and predictions based on a combination of both determinate and indeterminate tasks. In addition, after participants completed the reasoning task they were requested to recall their previously created mental models and then to decide whether a certain constellation of presented terms match. The data suggest that participants again first use the PMM rather than an alternative.

Method

Participants. Twenty-eight students (15 female, $M = 22.86$ years, $SD = 3.17$) participated in the study. Four participants were excluded from further analysis because they missed the chance level threshold of 21 correctly solved tasks on conclusion validation determined by a binomial test. Participants gave written consent and either received a small monetary reward or course credit for their participation.

Procedure, Materials, and Design. Participants were seated in front of a computer screen and first completed six training tasks. Using a within-participants design they then completed 48 balanced and randomized tasks chosen from a total of 288 tasks. There were 24 task sets each assigned only once. As is illustrated in Figure 1 trials exposed participants with (1) an initial premise processing, (2) a conclusion validation, and (3) a mental model validation phase.

Including instructions each session lasted approximately 45 to 60 minutes. Response times and answer correctness for both conclusion and mental model validation were logged. The material consisted of 144 determinate and 144 indeterminate tasks each presenting three consecutive premises (P1-P3) followed by one putative conclusion and a putative mental model. Premises and conclusions consisted of two terms. Terms were separated by a centered dot and each term either appeared to the left or to the right of it. The alignment of terms on the screen at the same time encoded the underlying

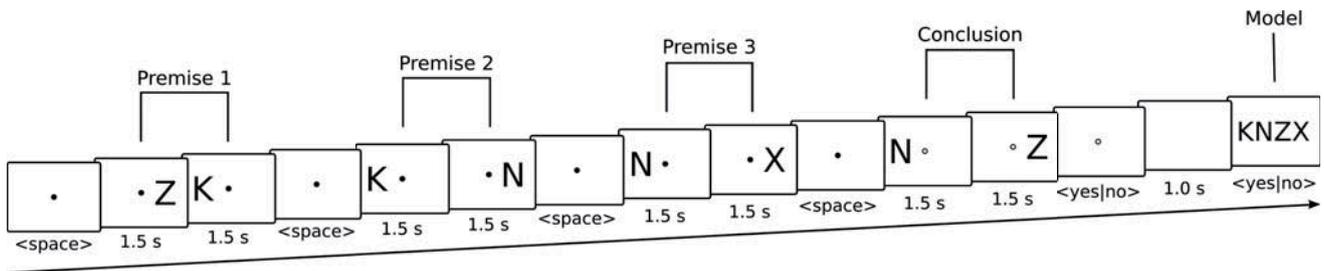


Figure 1: Sample trial. Spatial relations are indicated by their position, i.e. right aligned ‘Z’ in the first premise means ‘Z is to the right of.’ For each trial premise and conclusion terms were presented at least 1500 ms. To see the next pair of terms participants had to press the space bar in the premise processing phase. To complete the conclusion processing phase they had to press either the left or right arrow key representing ‘yes’ and ‘no’ respectively. Premise and conclusion presentation differed in the term separating center point. For the final model validation task participants had to respond using the arrow keys again.

ing spatial relation; if the first term appeared left aligned this indicated a “left-of” relation to the second term that consequently appeared right aligned as soon as the first was cleared from the screen and vice versa. Note that for illustrative purpose we use the terms “A,” “B,” “C,” and “D” whereas in the experiment, to avoid sequence effects or particular abbreviations that can be memorized, “K,” “N,” “X,” and “Z” were used (cf. Figure 1). Similarly, for each structurally equivalent premise across trials different terms were used resulting in 24 permutations of model terms. Table 1 shows premise combinations for determinate and indeterminate tasks. P1 and P2 differed in the presentation order of terms whereas P3 was the same in all trials.

After the premises were displayed a putative conclusion was presented in the same way. Merely the term separator differed indicating that the presented terms now belong to the conclusion. Eventually, after participants responded to the conclusion a configuration of four terms was displayed on the screen representing a putative mental model. In both determinate and indeterminate tasks participants had to decide whether the presented configuration matched one of the mental models they had previously constructed.

Cognitive Model

In general, configurations of spatially related terms are represented as mental model chunks with position slots holding the respective terms. Hence, premises, conclusions, or entire mental models share the same structure. Of the ACT-R internal parameters only activation noise (0.31), retrieval threshold (-0.75), and latency factor (0.4) deviated from the default values but were held constant across runs.

The ACT-R model processes the first premise without encountering any indeterminacy and integrates premise terms directly into an empty mental model chunk. It then makes no further use of the information encoded in the first premise. This is inspired by the findings of Mani and Johnson-Laird (1982) that human reasoners, having processed the first premise, forget the corresponding information; there is no need to remember it because there are no alternative men-

tal models possible that would later require a retrieval of this information to allow for a modification of the initial PMM. For successive premises, however, indeterminacy may occur due to multiple possible positions for term integration. Consequently, the ACT-R model creates extra representations for successive premises that allow for a later retrieval if indeterminacy occurs. The process of flagging a successive premise to make it available for later processing steps is referred to as “annotating a premise”. In particular, the ACT-R model flags the already inserted term after which indeterminacy occurred as the “reference object” term (RO) and the new term as the “to be located object” term (LO); the LO has to be inserted at the first free position rather than at the directly adjacent first fitting position that is already occupied by another term from a previously presented premise. Figure 2 illustrates the process of the construction of the initial PMM including annotation assignment.

If the PMM fails to validate the conclusion, the ACT-R model can now retrieve annotated premises and reuse them to modify the PMM to an alternative model as is illustrated in Figure 3. Annotation usage proceeds by transferring the current mental model chunk to the imaginal buffer and retrieving the previously annotated premise. An alternative term configuration can be obtained by moving the LO towards the RO, i.e. to the first fitting position. In a subsequent step the ACT-R model retrieves the conclusion chunk again and tests it in a second comparison step with the modified mental model. These steps are repeated until the presented conclusion agrees with a mental model causing the motor module to press the key for a positive response, or when it fails to retrieve further annotated premises resulting in a key press representing a negative response.

In the final mental model validation phase the ACT-R model first retrieves the initial UNI/PMM again and transfers it to the imaginal buffer. This transfer is necessary because, like the model construction phase illustrated in Figure 2, intermediate term retrievals repeatedly occupy the retrieval buffer. The reason for the explicit retrieval request for the UNI/PMM is that participants make fewer errors and need

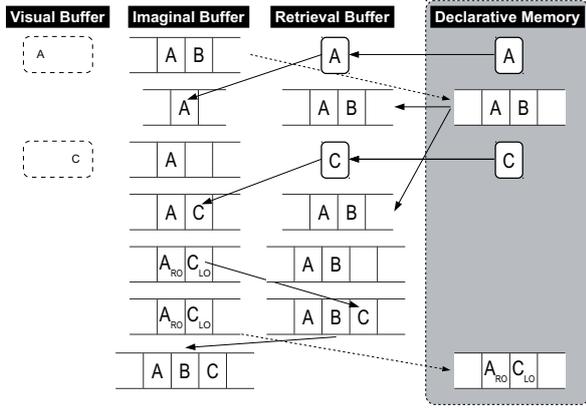


Figure 2: Preferred mental model (PMM) construction. This illustration starts with an incomplete mental model chunk in the imaginal buffer comprising just the information from the first premise. Objects processed by the visual module are linked to term chunks in declarative memory that the model needs to retrieve before it can integrate them into a mental model chunk. Note that the first premise is treated differently because the ACT-R model creates no explicit chunk for it; for simplicity, the corresponding processing steps are omitted here. Subscripts refer to the annotation flags “reference object” (RO) and “to be located object” (LO) necessary for identification in a potential later retrieval (cf. Figure 3).

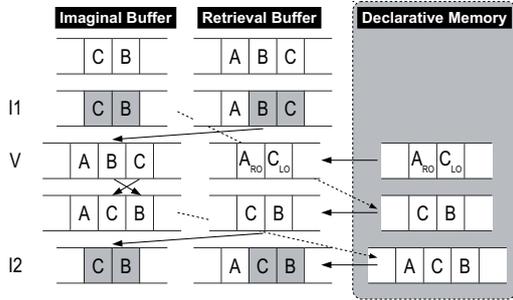


Figure 3: Annotation usage. Labels refer to mental model inspection (I2, I2) and variation (V). Comparisons across buffers are highlighted in gray. Note that this simplified illustration starts with a conclusion in the imaginal and a preferred mental model (PMM) in the retrieval buffer. If in I1 the comparison of conclusion and PMM fails, the PMM is transferred to the imaginal buffer, the conclusion is released to declarative memory, and an annotated premise is retrieved allowing for PMM variation. Eventually, the conclusion is retrieved again allowing for further comparisons in I2.

less time when no competing alternative model has to be used for validating the previous conclusion (cf. Figure 4b).

An important difference to the premise processing phase is that the ACT-R model skips the creation of an additional mental model chunk; instead, it compares the terms in the retrieval

buffer with the UNI/PMM in the imaginal buffer directly. Similarly, if in an indeterminate case the PMM fails to match, no modification takes place; instead, a retrieval request for an alternative model follows and eventually the comparison phase restarts. The motivation for this omission of additional mental model creation is that the respective terms are continuously displayed on the screen thus functioning as external memory; any memorization process involving the creation of internal representations would be redundant. In the premise processing phase, however, internal representations are necessary because no complete mental model is presented but has to be derived in multiple steps. Similarly, in the conclusion validation phase no external mental model representation is available. Without generating and keeping internal representations conclusion validation would, therefore, be impossible.

Results and Discussion

The results support the predictions of the PMMT regarding the preference for the PMM and the increasing reasoning difficulty with the transformation distance of alternative models (cf. Table 1). Table 2 shows the mean error rates (in %) and response times (in milliseconds) for both conclusion and mental model validation.

Table 2: Human data. Mean error rates (in %) and response times (in milliseconds) for both conclusion and mental model validation.

Type	Conclusion validation		Mental model validation	
	Error rate	RT	Error rate	RT
UNI	6.25	4379	7.33	1643
PMM	17.80	4813	15.18	1976
AM1	41.05	5688	47.92	2990
AM2	64.58	5402	67.71	3090

Pearson’s correlation was used to test whether human data correlated with model predictions. Correlations were computed on a by-task aggregate level for those tasks with an expected correct response resulting in 120 data points each for human and model means. The rationale behind this is that to correctly reject an invalid conclusion reasoners cannot prematurely terminate the validation process as they would have to test all possible mental models; premature termination is only possible with a correct mental model. Hence, if the task was to reject an invalid conclusion for both data and model predictions we expected no differences across conditions. Please also note that for mental model validation only those cases were subject to analysis in which the externally presented model matched the model that previously validated the conclusion.

Figures 4a and 4b compare data from 24 participants with predictions based on 2400 model runs. Each run processes one of the 24 task sets; hence, with each task set consisting of 48 tasks this resulted in a total of $2400 \times 48 = 115.200$ simulated experimental trials.

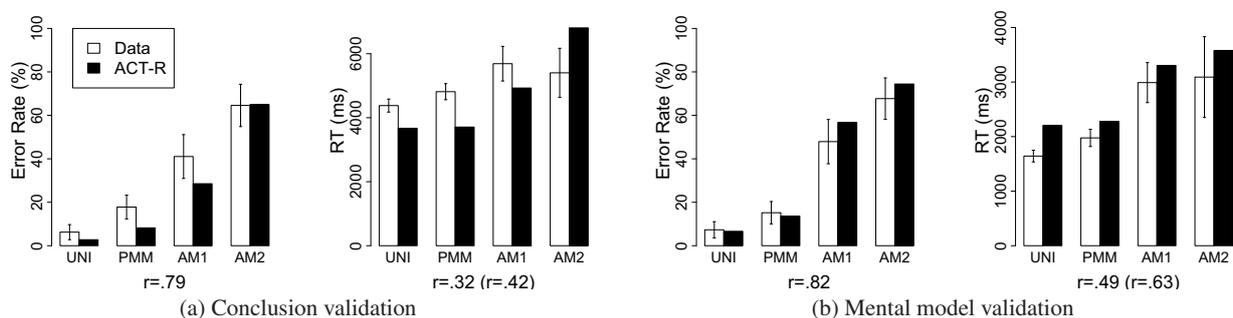


Figure 4: Average error rates and response times for conclusion and mental model validation. Conclusion validation either requires a mental model constructed on the basis of determinate premises (UNI) or indeterminate premises (PMM, AM1, AM2). Correlations were computed on a by-task aggregate level only for those tasks with an expected correct response resulting in 120 data points each for human and model means. Response time correlations were computed with and without taking the AM2 condition into account. For the latter, the corresponding coefficient is given in brackets. Error bars show 95% confidence intervals. All correlations were significant at a $p < .001$ level.

Error rate correlations between data and ACT-R predictions showed a significant effect for both conclusion, $r = .79, p < .001$ (cf. Figure 4a) and mental model validation, $r = .82, p < .001$ (cf. Figure 4b).

Response time correlations between data and ACT-R predictions showed a significant effect for both conclusion, $r = .51, p < .001$ (cf. Figure 4a) and mental model validation, $r = .33, p < .001$ (cf. Figure 4b). In the AM2 condition, however, response times for conclusion validation contradict the theoretical assumptions of the PMMT as there is an unexpected decrease in the average response time in comparison to the AM1 condition. However, an error rate of 64.58% results in only few contributing participants when only correct responses are taken into account. Furthermore, a lower response time in this context suggests that participants used the AM1 for conclusion validation rather than the predicted later stage AM2. As a last resort they may even have guessed. A similar explanation may hold for the average response time for mental model validation and a corresponding error rate of 67.71% being comparably high. Correlations, hence, were computed again without taking those tasks requiring an AM2 for conclusion validation into account based on 96 data points each for human and model means. Correlations then improved and showed a significant effect for both conclusion, $r = .61, p < .001$, and mental model validation, $r = .42, p < .001$.

When processing determinate premises with only one resulting mental model (UNI) participants were faster and made fewer errors than in the remaining conditions. When processing indeterminate premises a well-established preference effect could be reported for situations where only the preferred mental model (PMM) was necessary to validate a conclusion; participants were faster and made fewer errors than if an alternative model (AM1 or AM2) was required. They performed poorer, however, than in the UNI condition. According to

the PMMT this effect can be explained by additional processes that annotate indeterminate premises allowing for later PMM modifications to an alternative mental model. From an ACT-R theoretical perspective, because these processes require additional time, additional activation decay of the PMM also results; consequently, its retrieval at conclusion validation takes more time and is less reliable. Finally, a “distance effect” for AM1 and AM2 could be reported; the more modifications to the initial PMM were necessary, the less likely they were to be correctly accepted or rejected.

With respect to the mental model validation task human data suggest that differences in response times depend on the number of mental models created in the conclusion validation phase. In the single model condition (UNI/PMM) response times were lower than in the multiple model condition (AM1/AM2). No explicit differences between AM1 and AM2 emerged. For the AM2 condition this suggests that after the PMM reasoners tried only one of the two alternatives.

There are, however, slightly higher predicted response times for mental model validation in the AM2 condition (cf. Figure 4b) that could not be reported for the human data. From an ACT-R perspective there is a clear explanation: As suggested, the mental model validation process starts with the PMM generated in the premise processing phase. Accordingly, to successfully validate a conclusion in the AM1 condition, apart from the PMM chunk, the ACT-R model needed to create an additional chunk for the AM1. In the AM2 condition, however, it had to create chunks for both, the AM1 and the AM2. Hence, even if the same number of mental models are recalled in both conditions AM1 and AM2—as the response times in the human data suggest—in the AM2 condition the PMM experienced more activation decay because more time has passed since its last usage. Consequently, the mapping of activation to retrieval time results in higher values for the AM2 than for the AM1 condition.

Conclusion

Our starting point was the question of how human reasoners process determinate and indeterminate spatial descriptions and a corresponding cognitive model introduced previously (Ragni et al., 2010; Ragni & Brüssow, 2011). The advantage of the present study is that the stimulus material comprised simple visual presentations of term objects rather than complex linguistic presentation such as sentences. The rationale behind this is that interfering linguistic processes could thus be inhibited. In this respect, the present study bridges a gap between the fMRI study by Fangmeier et al. (2006) and Ragni et al. (2007): Fangmeier et al. investigated only determinate problems but presented the material in a plain and functional way, whereas Ragni et al. covered a wide range of indeterminate problems but presented premise information verbally.

Furthermore, we systematically investigated the processes of mental model variation—the transformation from an initial PMM to an alternative model—that so far has received only little attention. In that context the presented ACT-R model is both an algorithmic foundation of the PMMT and a theory explaining behavioral data obtained from experiments using either linguistic or pictorial representation of premise information. To further establish the model and to investigate the persistence of the outcome of the reasoning process, it was extended and tested with a wider range of tasks consistently presented non-linguistically. It gives detailed insights into why certain tasks are computationally more demanding than others and, apart from being informative about the reasoning process itself, it simulates an additional mental model validation task that is in line with the experimental results indicating a primacy effect for the PMM; the data suggest that reasoners start comparisons by recalling the PMM rather than an alternative. Future work will focus on this particular point as the question of how reasoners proceed, naturally, has implications for the ACT-R model itself. Currently, if the PMM fails to match an alternative model is requested. Restarting the variation phase by retrieving annotated premises, however, would also be a possible strategy.

Acknowledgments

This research was supported by the DFG (German National Research Foundation) in the Transregional Collaborative Research Center, SFB/TR 8 within project R8-[CSPACE]. The authors are grateful to Tobias Sonntag, Philip Stahl, and Andreas Klein for assistance in participants acquisition and experiment implementation.

References

- Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford University Press.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036-1060.
- Fangmeier, T., Knauff, M., Ruff, C. C., & Sloutsky, V. M. (2006). fMRI evidence for a three-stage model of deductive reasoning. *Journal of Cognitive Neuroscience*, *18*, 320-334.
- Jahn, G., Knauff, M., & Johnson-Laird, P. N. (2007). Preferred mental models in reasoning about spatial relations. *Memory & Cognition*, *35*, 2075-2087.
- Johnson-Laird, P. N. (1980). Mental models in cognitive science. *Cognitive Science*, *4*, 71-115.
- Johnson-Laird, P. N. (1983). *Mental models: Towards a cognitive science of language, inference, and consciousness*. Cambridge, MA: Harvard University Press.
- Knauff, M., Rauh, R., & Schlieder, C. (1995). Preferred mental models in qualitative spatial reasoning: A cognitive assessment of Allen's calculus. In *Proceedings of the 17th annual conference of the cognitive science society* (p. 200-205). Mahwah, NJ: Erlbaum.
- Mani, K., & Johnson-Laird, P. N. (1982). The mental representation of spatial descriptions. *Memory & Cognition*, *10*, 181-187.
- Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice Hall.
- Ragni, M., & Brüssow, S. (2011). Human spatial relational reasoning: Processing demands, representations, and cognitive model. In W. Burgard & D. Roth (Eds.), *Proceedings of the 25th AAAI conference on artificial intelligence*. San Francisco, CA: AAAI Press.
- Ragni, M., Fangmeier, T., & Brüssow, S. (2010). Deductive spatial reasoning: From neurological evidence to a cognitive model. In D. D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the 10th international conference on cognitive modeling* (p. 193-198). Philadelphia, PA: Drexel University.
- Ragni, M., Fangmeier, T., Webber, L., & Knauff, M. (2007). Preferred mental models: How and why they are so important in human reasoning with spatial relations. In C. Freksa, M. Knauff, B. Krieg-Brückner, B. Nebel, & T. Barkowsky (Eds.), *Spatial cognition V: Reasoning, action, interaction* (pp. 175-190). Berlin: Springer.
- Ragni, M., Knauff, M., & Nebel, B. (2005). A computational model for spatial reasoning with mental models. In B. Bara, B. Barsalou, & M. Bucciarelli (Eds.), *Proceedings of the 27th annual conference of the cognitive science society* (p. 1064-70). Mahwah, NJ: Erlbaum.
- Rauh, R., Hagen, C., Knauff, M., Kuß, T., Schlieder, C., & Strube, G. (2005). Preferred and alternative mental models in spatial reasoning. *Spatial Cognition and Computation*, *5*, 239-269.
- Rauh, R., Hagen, C., Schlieder, C., Strube, G., & Knauff, M. (2000). Searching for alternatives in spatial reasoning: Local transformations and beyond. In L. R. Gleitman & A. K. Joshi (Eds.), *Proceedings of the 22nd annual conference of the cognitive science society* (p. 871-876). Mahwah, NJ: Erlbaum.

A multinomial model of applying recognition to judge between multiple alternatives.

C. Philip Beaman (c.p.beaman@reading.ac.uk)

School of Psychology & Clinical Language Sciences, University of Reading

Earley Gate, Whiteknights, Reading RG6 6AL, UK

Abstract

Proponents of the “fast and frugal” approach to decision-making suggest that inferential judgments are best made on the basis of limited information. For example, if only one of two cities is recognized and the task is to judge which city has the larger population, the *recognition heuristic* states that the recognized city should be selected. In preference choices with >2 options, it is also standard to assume that a “consideration set”, based upon some simple criterion, is established to reduce the options available. A multinomial processing tree model is outlined which provides the basis for estimating the extent to which recognition is used as a criterion in establishing a consideration set for inferential judgments.

Keywords: Decision-making; n-AFC; MPT models; Recognition heuristic.

Introduction

A much studied approach to the problem of determining how individuals (or groups) judge between two options is the “fast and frugal” heuristic framework advocated by Gigerenzer and colleagues (e.g., Gigerenzer, Hertwig & Pachur, 2011). This framework aims to provide both descriptive and normative accounts of how judgments are made in the real world, where time and computational resources may preclude optimizing strategies (e.g., multiple regression) where all possible sources of information are consulted and appropriately weighted combinations of these are applied to inform the judgments made.

One of the most basic principles underlying the fast and frugal approach is the *Recognition Principle*, later formalized as the Recognition Heuristic (RH; Goldstein & Gigerenzer, 2002) and implemented in ACT-R (Marewski, & Mehlhorn, 2011; Schooler & Hertwig, 2005). The RH acts as a stopping rule for more complex heuristics (e.g., Take-the-Best, Gigerenzer & Goldstein, 1996) and states simply that, “If the task is to judge which of two options scores highest on a given criterion, and only one of the two options is recognized, infer that the recognized option scores highest on the criterion”. For example, the heuristic might be used to judge which of two cities has the larger population (Goldstein & Gigerenzer, 2002), which of two individuals is the wealthiest (Frosch, Beaman & McCloy, 2007) or which of two diseases occurs most frequently per year (Pachur & Hertwig, 2006).

The heuristic is applied when the criterion in question is known to correlate positively with the probability of recognition (Volz, Schooler, Schubotz, Raab Gigerenzer & von Cramon, 2006), although the reverse inference can also be made in the unlikely event that the correlation is known to be negative (Goldstein & Gigerenzer, 2002). In either

case, the heuristic takes advantage of information latent in the environment to inform inference and minimize the amount of knowledge sought or retrieved from memory (or other, external sources) about the items in question prior to making a judgment.

The RH has not been universally accepted as descriptive account of human inference, however. Hilbig and colleagues, in particular, have drawn attention to the difficulty in identifying when recognition *per se* is used to inform judgment rather than knowledge which accords with recognition (Hilbig, 2010; Hilbig, Erdfelder & Pohl, 2010; Hilbig & Pohl, 2008; Hilbig, Pohl & Bröder, 2009; Hilbig & Richter, 2011). Given the positive correlation between the probability of recognition and the criterion in question, it is inevitable that many of the other cues that could in principle be consulted (even if only one of the items is recognized) will result in the same inference as recognition alone.

One way of addressing this problem is applying a multinomial processing tree (MPT) model (e.g., Hilbig et al., 2010). MPT models assume sequential, independent operations (akin to additive factors logic; Sternberg, 1998) which can be expressed as a tree structure, with alternative processes at each branch point each associated with a parameter that represents the probability of traversing that particular branch. The tree structure terminates with an observable outcome, and the models are compared to the data by estimating the best-fitting parameters and comparing the frequency counts of each outcome in experimental data to the expected outcomes given the parameters estimated (for formal reviews, see Bachelder & Rifer, 1999).

As a concrete example, consider the position in a two-alternative forced choice task (2-AFC) where the subject is asked to indicate which of the two options scores highest on the criterion of interest. Here, three trees need to be drawn. In the scenario described by the first tree, the subject recognizes neither of the options and is forced to guess between them. There are therefore only two possible outcomes, a correct guess or an incorrect guess, and hence only two branches to the tree. The probability of a correct guess (the branch leading to the frequency count of correct answers) is associated with the parameter g , and the probability of an incorrect guess is therefore given by $(1-g)$. Where guessing is truly random, $g=.5$. The scenario described by the *third* tree is similarly of little direct interest to the study of the RH. Here, the subject recognizes both options and uses knowledge to choose between them, with probability correct given by parameter b (“*knowledge validity*”) and the same two-branch tree-structure as previously. This tree is potentially open to expansion,

assuming that suitable candidate knowledge processes can be identified. Of more immediate interest is the *second* decision tree, which describes the scenario in which exactly one of the two options is recognized. Here, there are four possible outcomes: the recognized item might be chosen or not, and the choice might be correct or incorrect. There are also multiple ways in achieving these outcomes.

Suppose the RH is applied uncritically, then the probability that the RH produces the correct answer is given by parameter a (“*recognition validity*”). However, whether the RH is applied uncritically is itself governed by a parameter, r . Thus, the probability the RH is applied and is correct is given by $r.a$ and the probability the RH is applied but results in an incorrect inference is given by $r(1-a)$. If the RH is not applied uncritically, then the inference that the recognized item is correct might be made by the alternative route of applying knowledge. The probability that this occurs is the joint probability of the RH being valid, knowledge being valid, and knowledge being used, given by $(1-r).b.a$. The routes by which decisions might be made according to this MPT model are given exhaustively in Table 1.

Table 1: Probabilities of choosing correctly or incorrectly via different routes of three simple decision trees. The probability of arriving at the end point of any particular branch of a decision-tree is the product of the parameters associated with each branch-point.

Recognize 0: g (correct)
$(1-g)$: (incorrect)
Recognize 1: $r.a$ (choose recognized, correct)
$r(1-a)$ (choose recognized, incorrect)
$(1-r)b.a$ (choose recognized, correct)
$(1-r)b(1-a)$ (choose unrecognized, correct)
$(1-r)(1-b)a$ (choose unrecognized, incorrect)
$(1-r)(1-b)(1-a)$ (choose recognized, incorrect)
Recognize 2: b (correct)
$(1-b)$ (incorrect)

From this table, it is clear that MPT models are well-suited to answering the question of the extent to which the RH is employed in any given 2-AFC inference task. Estimation the best-fitting set of parameters from the data provides the most likely rate of application of the RH in terms of the r parameter. This can be done across the sample of all subjects within the experiment or, where a sufficiently large data-set is available for estimations to be made on an individual basis, it can be done per subject. This may be useful if, for example, r and a are not independent for some subjects (see, e.g., Beaman et al., 2010). However, because of the limited data available, in the study reported below only estimation across the full dataset was attempted.

Multi-Alternative Inferences

Not all inferences of the kind described above involve choosing between only two alternatives. Frosch et al. (2007) presented 3- or 4-alternatives to choose from in a wealth judgment task. Marwesi et al. (2010) likewise examined the incidence of RH use when judging likelihood of electoral success amongst named politicians. Theoretical analyses of the success of the RH as a normative theory under these circumstances are also provided by McCloy, Beaman & Smith (2008). Both *a priori* theoretical analyses and empirical studies rely upon the presumption that there are independent processing stages in which N alternative options are reduced to a smaller number from which to choose, but the decision-making processes employed to reduce the number of alternative options may differ from those employed to choose from amongst those options. This situation parallels one in the literature on consumer choice, in which it is assumed that a “consideration set” is initially formed from within all the options available (Howard, 1963; Wright & Barbour, 1977). Indeed, in some circumstances, criterion-inference judgments are equivalent to preference choices (e.g., if subjects are asked to indicate which politician they believe will win a given election, as in Marewski et al.’s (2010) data, they are *de facto* being asked which candidate they believe the public as a whole will prefer to win the election). These circumstances also lend themselves to modeling by means of MPTs. One difference between criterion-inference and preference-choice, may lie in the way in which criteria are applied for preference-choice. However, this does not preclude the possibility that consideration sets may be invoked in multi-alternative inference, even if the nature of the criteria applied may need not be identical across the two situations.

The structure of 3-AFC judgments.

To model such multi-alternative inferences using MPTs, an appropriate data-set is required and tree structures representing the structure of the model must be constructed. Here, the example of 3-AFC inference is explored using data from Frosch et al. (2007) in a study in which subjects were asked to indicate which of three names taken from the *Sunday Times* rich list (an annual compilation of the richest individuals in the UK) was the wealthiest. 27 such names were chosen and, from within this group, were randomly 22 lists of three names each were produced by random selection. 26 subjects were then each given the same series of three names to choose from per trial and were asked to indicate which name they thought was the wealthiest. They were then re-presented with all the names and asked to indicate which names they had known prior to the beginning of the experiment. As with 2-AFC, the trees describing scenarios in which either all the options or none of the options are recognized are relatively uninteresting, being governed purely by single guessing or knowledge parameters (g and b , respectively).

For 3-AFC scenarios in which one item is recognized, a plausible tree structure is given in Figure 1. The tree

structure when one item from three is recognized differs from the corresponding structure when one item is recognized in a 2-AFC task. As shown in Figure 1, there is an extra step with 3-AFC when the RH is not applied, knowledge is valid (i.e., will provide the correct inference) but the RH is not valid (will provide an incorrect inference). The reasons for this can readily be discerned as follows: If knowledge is inaccurate and is used in preference to the RH, then it will always lead to a false choice, either of the recognized item (if recognition is also an invalid cue) or one of the unrecognized items (if recognition is a valid cue), as in the 2-AFC situation. However, if knowledge is valid then this knowledge must be either that the one recognized item is high on the criteria or that the recognized item is low on the criteria. If the former, the recognized item will be chosen, if the latter then it will be excluded from further analysis. If both knowledge and recognition provide valid cues, then the recognized item must be the correct item and it will be chosen. If, however, recognition is not a valid cue then the correct item must be one of the two unrecognized options: hence there is an extra guessing stage to choose between these two. In terms of the equations presented in Table 1, the conditional probability of guessing correctly when following this particular branch of the tree can be expressed as, $(1-r_1) \cdot b_1 \cdot (1-a_1) \cdot g_1$,

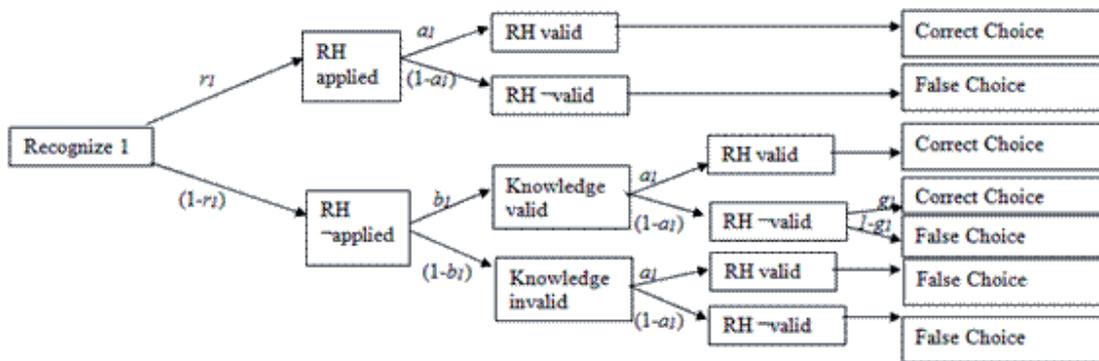


Figure 1. A 3-AFC decision tree indicating possible processing paths leading to correct or incorrect criterion inferences when exactly one item from three is recognized.

A similar logic can be applied to construct the tree structure for scenarios where two of the three options were recognized. The decision tree for the scenario when two items (out of three) are recognized is more complex than when only one item is recognized (Figure 2). Here, the *RH inclusion rule* is explicitly distinguished from the RH itself. The RH inclusion rule operates to winnow down the number of options under active consideration to a consideration set defined in terms of whether the items were recognized or not. Thus the RH inclusion rule operates along the recognition principle, as does the RH, but unlike the RH is

only the first step in establishing – by a longer chain of inference – which of the items scores most highly on the criterion. It is an empirical question whether the RH inclusion rule and the RH are applied at equivalent rates amongst the same group of subjects attempting the same criterion judgment task.

Figures 1 and 2 show how inferences may be made on the basis of recognition, knowledge, or recognition plus knowledge when three items are presented but recognition is incomplete. Under these circumstances, of course, the chosen item may also be either recognized or unrecognized and tracing the relevant branch of the decision tree gives this information also (e.g., if valid knowledge is used and recognition is also a valid cue then the correct choice made must be the choice of the recognized option).

When the inclusion rule is applied, recognition may be a valid or an invalid cue. If it is invalid, then straightforwardly, the choice of item must be incorrect as only the two recognized items are under consideration. If, however, recognition is a valid cue then correct inference is still not guaranteed as a choice must be made between the two recognized items. If the inclusion rule is not applied, then the knowledge about the two recognized items consulted instead may also be either valid or invalid. If it is valid and recognition is also a valid cue, that is to say the

correct item is one of the two recognized items, then a correct inference is made because valid knowledge of the two items is sufficient to choose between them. If the inclusion rule is invalid, then valid knowledge must be of a different type – knowledge that neither of the recognized items is correct. In this case too, however, correct inference is guaranteed. Following the same lines of argument, if knowledge is invalid then an incorrect choice is made regardless of whether recognition is valid; that is, regardless of whether the invalid choice is of a recognized or unrecognized item.

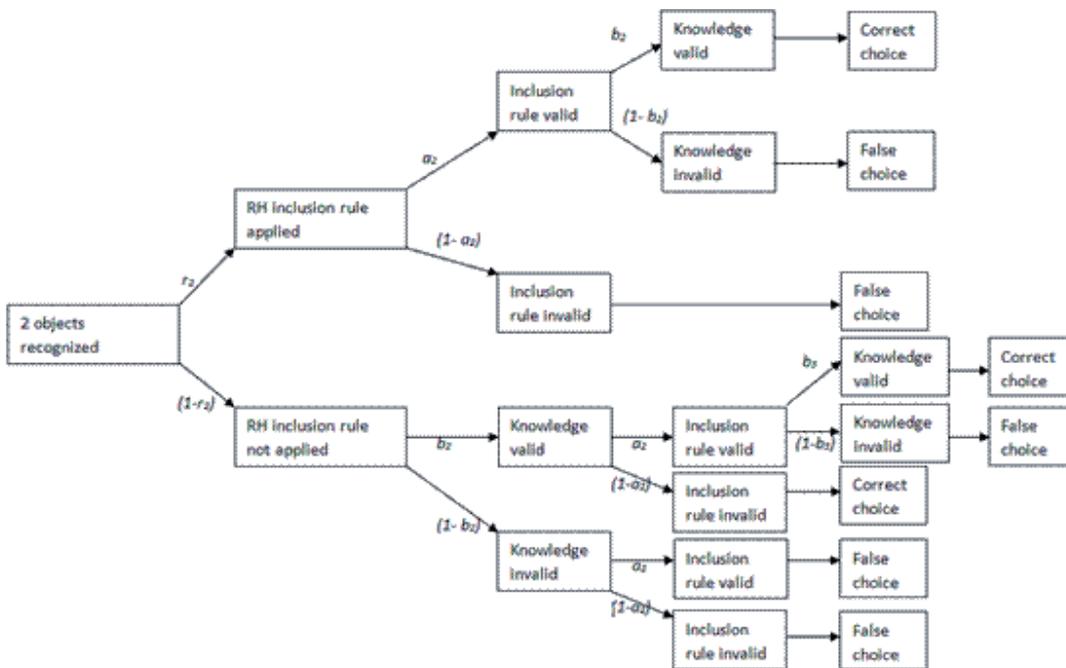


Figure 2. A 3-AFC decision tree indicating possible processing paths leading to correct or incorrect criterion inferences when exactly two items from three are recognized

Comparing the Model to “Rich List” Data

The best-fitting parameters for the model were estimated from Frosch et al.’s (2007) 3-AFC study using MultiTree software (Moshagen, 2010). “Guessing” parameters for situations in which neither knowledge nor recognition could be employed were set to produce chance-level accuracy (i.e., $g = .33$, $g_1 = 0.5$).

An initial model assumed a single value for knowledge validity to determine – based on knowledge – whether one of two recognized items was correct and also *which* of these two recognized items was the correct option (i.e., on Figure 2, $b_3 = 1.0$). For this model, recognition and knowledge validities were separately estimated for each recognition scenario (recognize 0, 1, 2 or 3) as there are reasons to suppose that knowledge validity should vary as a function of the number of items recognized (e.g., Smith, Beaman & McCloy, 2011). Estimating all parameters other than the “guessing” parameters, g and g_1 , a goodness-of-fit test shows that the best-fitting version of this model nevertheless produced expected results which differed significantly from those observed, $G^2=11.01$, $df=3$, $p = .01$.

A second model, assuming a decision-point (with validity b_2) that one of the recognized items – based on knowledge – was the correct item and a further decision-point (again, based on knowledge but with a different validity, b_3) produced expected results that did not differ significantly from those observed, $G^2=1.05$, $df=2$, $p=.59$.

Knowledge validity when all three items were recognized was estimated as 0.47 and when one item of the three was recognized knowledge validity was 0.56. Knowledge validity for correctly realizing –based on knowledge – that one of the two recognized items was the correct choice was high, at 0.91 but choosing between these two items once this decision was made was difficult, with validity of choices estimated at 0.30.

Interestingly, for this model, the probabilities r_1 and r_2 that the RH and the RH-inclusion rule were employed (Figures 1 and 2) were estimated as .64 and .63 respectively, and the validities of these two rules were also estimated as being very similar, a_1 (RH validity) = .72 and a_2 (inclusion rule validity) = .73. Constraining these two rules and their validities to take the same values resulted in a model where $a = .73$ and $r = .63$ which did not differ significantly from the baseline model, $\Delta G^2 = .19$, $df = 2$, $p = .91$, nor from the data, $G^2=1.25$, $df=4$, $p=.88$. For the 3-AFC situation, therefore, it appears that there is no discernible difference between using the RH to guide choice when only one item is recognized and using the RH-inclusion rule to form a consideration set which is subsequently informed by knowledge. Full parameter values for this version of the model are given in the appendix, a comparison of the model to the data in terms of observed and expected frequencies with which recognized and unrecognized items were chosen either correctly or incorrectly is given in Table 2. Although

parameter overfitting is a legitimate concern in any modelling domain, on which may be fuelled by observation of the close fits between expected and obtained results here, it is worth noting that – like standard hypothesis testing – a failure to provide a good fit given certain assumptions (e.g. that $r=1.0$) may be as informative as obtaining a good fit, and it is on this basis that the G^2 goodness-of-fit statistic operates.

Table 2: The frequency of responses tabulated according to whether they were correct or not, and whether the option chosen was recognized or not. Observed responses are taken directly from the data of Frosch et al. (2007), expected responses are the frequencies of responses within those particular categories given a model with the tree structures shown in Figures 1 and 2 and the parameters given in the appendix.

	Frequency of Responses	
	<i>Observed</i>	<i>Expected</i>
Recognize 0:		
Correct	40	35
Incorrect	65	70
Recognize 1:		
Correct (recog)	83	84.40
Incorrect (recog)	25	23.69
Correct (\neg recog)	10	9.87
Incorrect (\neg recog)	16	16.19
Recognize 2:		
Correct (recog)	79	77.81
Incorrect (recog)	62	62.79
Correct (\neg recog)	14	14.32
Incorrect (\neg recog)	4	4
Recognize 3:		
Correct	81	81
Incorrect	93	93

Finally, models in which the use of the RH was set at the level of RH use indicated by the data, if choice of the recognized item was taken as an index of RH-use (i.e., $r_r=.78$), differed significantly from the best-fitting model, $\Delta G^2=5.15$, $df=1$, $p=.02$, but not from the data, $G^2=6.2$, $df=3$, $p=.1$. However, if the same exercise is repeated for the RH-inclusion rule, the resulting model differs both from the best-fitting model, $\Delta G^2=13.59$, $df=1$, $p=.0002$, and from the data, $G^2=14.65$, $df=3$, $p=.002$. Using RH-adherence rate as an index of the RH-inclusion rule is therefore likely to be misleading.

Discussion

The modeling results reported here indicate the viability of applying the recognition principle as a means of forming a “consideration set” in criterion judgment as well as preference judgment, when there are more than two alternatives to consider. The particular tree structures proposed for an independent series of decisions can also be

fit to the data with estimated parameters providing expected results that do not differ significantly from those observed in the data (Table 2). Although the particular data-set modeled here (relative wealth judgments) clearly require a criterion judgment, the distinction between criterion and preference judgments is less distinct with other stimuli. For example, Marewski et al. (2010) apply the RH to understanding judgments about the relative success of candidates in German elections. Although this is clearly a criterion judgment (“who is most likely to win?”), it can also be framed as a preference judgment (“who is the best candidate?”) whilst still potentially meeting the prerequisites for applying the recognition heuristic or recognition principle (“infer that the recognized option scores highest on the criterion”). Elucidating the means by which multi-alternative inferences may be made, even if the expansion is the relatively modest one from 2-AFC to 3-AFC, helps indicate where there are possible similarities between preference choice (where multiple alternatives are frequently presented) and criterion inference (where, typically, paired choices are presented, as in the “city size” task popularized by Gigerenzer and Goldstein (1996).

One interesting feature that became apparent when modeling these data was the need for multiple different parameters to represent knowledge validity, if not recognition validity. This both supports Smith et al.’s (2011) arguments that knowledge validity should vary as a function of the number of items recognized (see also Beaman et al., 2010) and highlights the need for different types of knowledge to be consulted to choose that the correct item is one of the recognized set (e.g., b_2) and which member of the recognized set the correct item may be (e.g., b_3 ; $b_2 \neq b_3$).

The model also indicates that, at least for the data-set considered here, there may be little or no difference between the application and validity of a “recognition inclusion rule” and the recognition heuristic *per se*. This is something of a surprise as, *a priori*, one would assume that a recognition inclusion rule might be employed, as a cognitive short-cut, even more frequently than the RH itself. However, it may simply be that the 3-AFC was too similar to 2-AFC for any differences to become apparent. The data also show that recognition adherence rate is not a good measure of use of the recognition principle when more than one item is recognized and a recognized item is chosen. The resulting model, where r_2 is set at the adherence rate, differs significantly from both the baseline, best-fitting model and the data. Setting r_1 , the incidence of employing the RH, also produces a model that differs significantly from the baseline model although, in this case, both models fit the data.

Finally, it is worth noting some limitations of the current modelling approach. Like all formal approaches, it relies upon a number of background assumptions. The need for multiple parameters and the relative independence of certain parameters have already been alluded to, but of equal concern is the lack of an underlying process model. The framework as it currently exists blurs the boundaries

between the existence of cognitive processes and their efficacy, and the challenge is to provide a process model which both draws clear distinctions between the two and shows how, and when, knowledge is consulted.

Acknowledgments

The multinomial modeling research reported here was supported by ESRC grant RES-062-23-1752. The experimental data to which the model was fit was funded by the Leverhulme Trust, grant no. F/00 239/U. I am grateful to Caren A. Frosch for collecting and analyzing these data and to Philip T. Smith for many useful discussions.

References

Batchelder, W. H., & Riefer, D. M. (1999). Theoretical and empirical review of multinomial process tree modeling. *Psychonomic Bulletin & Review*, *6*, 57-86.

Beaman, C. P., Smith, P. T., Frosch, C. A., & McCloy, R. (2010). Less-is-more effects without the recognition heuristic. *Judgment & Decision-Making*, *5*, 258-271.

Frosch, C., Beaman, C. P., & McCloy, R. (2007). A little learning is a dangerous thing: An experimental demonstration of ignorance-driven inference. *Quarterly Journal of Experimental Psychology*, *60*, 1329-1336

Gigerenzer, G., & Goldstein, D. G. (1996). Reasoning the fast and frugal way: Models of bounded rationality. *Psychological Review*, *103*, 650-669.

Gigerenzer, G., Hertwig, R., & Pachur, T. (2011). *Heuristics: The foundations of adaptive behavior*. Oxford: OUP.

Goldstein, D., & Gigerenzer, G. (2002). Models of ecological rationality: The recognition heuristic. *Psychological Review*, *103*, 650-669.

Hilbig, B. E. (2010). Precise models deserve precise measures: a methodological dissection. *Judgment and Decision Making*, *5*, 272-284

Hilbig, B. E., Erdfelder, E., & Pohl, R. F. (2010). One-reason decision-making unveiled: A measurement model of the recognition heuristic. *Journal of Experimental Psychology: Learning, Memory & Cognition* *36*, 123-134

Hilbig, B. E. & Pohl, R. F. (2008). Recognizing users of the recognition heuristic. *Experimental Psychology*, *55*, 394-401

Hilbig, B. E., Pohl, R. F. & Bröder, A. (2009). Criterion knowledge: A moderator of using the recognition heuristic? *Journal of Behavioral Decision Making*, *22*, 510-522

Hilbig, B. E., & Richter, T. (2011). Homo heuristicus outnumbered: Comment on Gigerenzer and Brighton (2009). *Topics in Cognitive Science*, *3*, 187-196

Howard, J. A. (1963). *Consumer Behaviour: Application of theory*. New York: McGraw-Hill.

Moshagen, M. (2010). multiTree: A computer program for the analysis of multinomial processing tree models. *Behavior Research Methods*, *42*, 42-54.

Marewski, J. N., Gaissmaier, W., Schooler, L. J., Goldstein, D. G., & Gigerenzer, G. (2010). From recognition to

decisions: Extending and testing recognition-based models for multi-alternative inference. *Psychonomic Bulletin & Review*, *17*, 287-309.

Marewski, J. N., & Mehlhorn, K. (2011). Using the ACT-R architecture to specify 39 quantitative process models of decision making. *Judgment and Decision Making*, *6*, 439-519

McCloy, R., Beaman, C. P., & Smith, P. T. (2008). The relative success of recognition-based inference in multi-choice decisions. *Cognitive Science*, *32*, 1037-1048

Pachur, T. & Hertwig, R. (2006). On the psychology of the recognition heuristic: Retrieval primacy as a key determinant of its use. *Journal of Experimental Psychology: Learning, Memory & Cognition*, *32*, 983-1002.

Schooler, L. J., & Hertwig, R. (2005). How forgetting aids heuristic inference. *Psychological Review*, *112*, 610-628.

Smith, P. T., Beaman, C. P., & McCloy, R. (2011). *The recognition heuristic and knowledge use*. Poster presented to Subjective Probability, Utility & Decision-Making (SPUDM) 23, Kingston-upon-Thames, London.

Sternberg, S. (1998). Discovering mental processing stages: The method of additive factors. In: D. Scarborough & S. Sternberg (Ed.s) *An invitation to cognitive science, volume 4: Methods, models and conceptual issues*. Cambridge, Ma.: MIT Press.

Wright, P. & Barbour, T. (1977). Phased decision strategies: Sequels to initial screening. In: M. Starr & M. Zeleny (Ed.s) *Multiple criteria decision making*. Amsterdam: North Holland Publishing.

Appendix

Parameter values for the version of the 3-AFC model fit to the data in Table 2. Standard errors for all of the estimates are given in parentheses.

Free Parameters:	
Probability applying recognition principle	
r_1 (RH) = r_2 (inclusion rule)	.63 (.05)
Recognition validity for RH and recognition inclusion rule	
$a_1 = a_2$.73 (.03)
Knowledge validity	
b_1 (1 item recognized)	.56 (.11)
b_2 (2 items recognized)	.91 (.05)
b_3 (choice between 2 recognized items)	.47 (.04)
b_4 (3 items recognized)	.30 (.17)
Fixed Parameters:	
Probability correct guess	
g	.33
g_1	.5

Data Acquisition Dynamics and Hypothesis Generation

Nicholas D. Lange (ndlange@gmail.com) †‡

Rick P. Thomas (Rickey.P.Thomas-1@ou.edu) ‡

Eddy J. Davelaar (e.davelaar@bbk.ac.uk) †

†Birkbeck, University of London, Department of Psychological Sciences,
Malet Street, London, UK, WC1E 7HX

‡University of Oklahoma, Department of Psychology
Lindsey Street, Norman, OK, USA 73019

Abstract

When formulating explanations for the events we witness in the world temporal dynamics govern the hypotheses we generate. In our view, temporal dynamics influence beliefs over three stages: data acquisition, hypothesis generation, and hypothesis maintenance and updating. This paper presents experimental and computational evidence for the influence of temporal dynamics on hypothesis generation through dynamic working memory processes during data acquisition. Results suggest that data acquired from the environment undergo dynamic competition in working memory, the results of which dictate the weights allocated to individual data in the generation process.

Keywords: hypothesis generation, temporal dynamics, working memory, abduction, diagnostic reasoning

Introduction

Hypothesis generation is a pre-decisional process by which we formulate explanations and beliefs regarding the occurrences we observe in our environment. The hypotheses we generate from long-term memory bring structure to many of the ill-structured decision making tasks we encounter on a daily basis. As such, hypothesis generation represents one of our most fundamental and ubiquitous cognitive faculties. Given such regularity, it is no surprise that hypothesis generation forms a core component of several professions. Auditors, for instance, must generate hypotheses regarding abnormal financial patterns and mechanics must generate hypotheses concerning car problems. Perhaps the clearest example, however, is that of medical diagnosis. A physician observes a pattern of symptoms presented by a patient (i.e., data) and uses this information to generate likely diagnoses (i.e., hypotheses) in an effort to explain the patient's current disease state. Given these examples, the importance of developing a full understanding of the processes underlying hypothesis generation is clear, as the consequences of impoverished or inaccurate hypothesis generation can be injurious.

When engaged in hypothesis generation tasks, cognitive limitations place constraints on the acquisition of bits of data used to cue long-term memory for the retrieval of likely hypotheses. Important to the present work is the fact that data acquisition most often occurs serially. This, in turn, dictates that individual pieces of data are acquired in some temporal relation to one another. These constraints, individual data acquisition over time and the relative ordering of data, are likely to have significant consequences

for hypothesis generation processes. Given these basic constraints it is intuitive that temporal dynamics must form an integral part of any comprehensive account of hypothesis generation processes. In our view temporal dynamics influence beliefs over three stages: data acquisition, hypothesis generation, and hypothesis maintenance and updating (as further data is acquired or judgments and decisions rendered). This paper concerns the temporal dynamics unfolding over the initial data acquisition phase which until now has remained unaddressed.

At present there exists limited data concerning the temporal dynamics of hypothesis generation tasks. Thus, the influences of the constraints operating over these processes are not yet well understood. Until such influences are addressed at an empirical and theoretical level a full understanding of hypothesis generation processes will remain speculative. Interest in understanding these underlying temporal dynamics is increasing however. For instance, Sprenger & Dougherty (2011) found a general recency bias in hypothesis generation whereby people tended to generate hypotheses more consistent with data received later than data received earlier. Additionally, Mehlhorn et al. (2011) investigated how hypotheses' memory activations are influenced by the amount of data that has been received at various time steps finding increases in memory activation with increases in supporting data.

HyGene (Dougherty, Thomas, & Lange, 2010; Thomas, Dougherty, Harbison, & Sprenger, 2008), short for hypothesis generation, is a computational architecture addressing hypothesis generation, evaluation, and testing. This framework has provided a useful account through which to understand the cognitive mechanisms underlying these processes. Here we extend this work by incorporating working memory dynamics from the context activation model of list memory (Davelaar, et al., 2005) to account for data acquisition dynamics subserving the cued recall process inherent in hypothesis generation.

HyGene & Temporal Dynamics

HyGene rests upon three core principles. First, it is assumed that hypothesis generation represents a generalized case of cued recall. Data observed in the environment (D_{obs}), which one would like to explain, act as cues prompting the retrieval of hypotheses from long-term memory (LTM). For

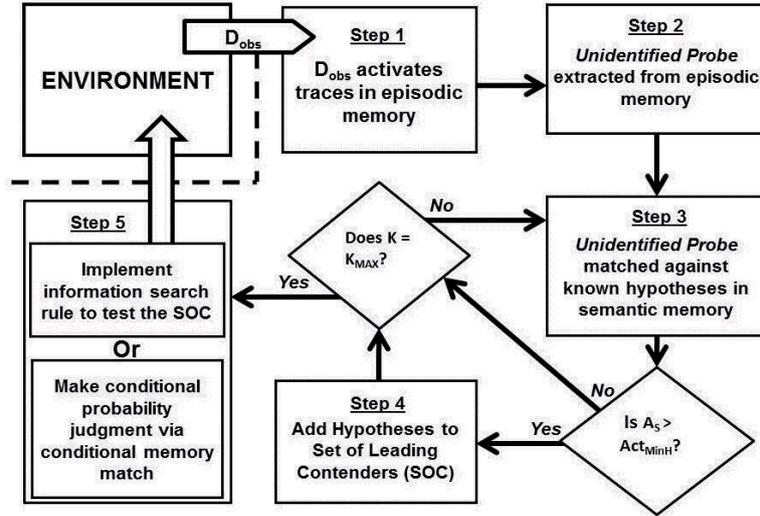


Figure 1: Flow diagram of processing in HyGene

instance, when a physician examines a patient, he/she uses the symptoms expressed by the patient as cues to related experiences stored in LTM. These cues activate a subset of related memories in episodic memory which guide the generation of hypotheses from semantic memory. These retrieval processes are indicated in steps one, two, and three of Figure 1. As viable hypotheses are retrieved from LTM they are placed in the Set of Leading Contenders (SOC) as demonstrated in step four. The SOC represents HyGene's working memory construct to which the second principle applies.

The second principle holds that the quantity of hypotheses that can be maintained at one time is constrained by cognitive limitations as well as task characteristics. That is, the more working memory resources that one has available to devote to the generation and maintenance of hypotheses, the more accommodating the SOC will be of additional hypotheses. Working memory capacity places an upper bound on the amount of hypotheses (and data) that one will be able to maintain at any point in time. In many circumstances, however, attention will be divided by a secondary task. Under such conditions this upper bound is reduced as the alternative task siphons resource that would otherwise allow the population of the SOC to its unencumbered capacity (Dougherty & Hunter, 2003a; Dougherty & Hunter, 2003b; Sprenger & Dougherty, 2006; Sprenger et. al., 2011).

The third principle states that the hypotheses maintained in the SOC form the basis from which probability judgments are derived and provide the frame from which hypothesis testing is implemented. This principle underscores the function of hypothesis generation as a pre-decisional process underlying higher-level decision making tasks and can be seen as step five in the diagram.

These assumptions form the core of HyGene's theoretical framework. HyGene in its current form is *static* with regards to data acquisition and utilization. The model receives all

available data from the environment simultaneously and engages in only a single iteration of hypothesis generation. Given the static nature of the model, each piece of data used to cue LTM contributes equally to the recall process. There is reason to suspect, however, that all available data do not generally contribute equally. What is needed is an understanding of working memory dynamics as data acquisition, hypothesis generation, and maintenance processes unfold and evolve over time in hypothesis generation tasks.

A Dynamic Model of Data Acquisition and Hypothesis Generation

We now forward a dynamic version of HyGene in which the activations of individual pieces of data acquired from the environment fluctuate over time in working memory prior to hypotheses being generated from long-term memory. The activation levels possessed by each piece of data at the time of generation are used as weights in the retrieval of hypotheses. This allows the activation of each piece of data in working memory to govern its individual contribution to the generation process. These dynamic working memory processes were borrowed from the context-activation model of memory (Davelaar et al., 2005). This model dictates that the activations of the items in working memory systematically fluctuate over time as the result of competing processes described by Equation 1.

$$x_j(t+1) = \lambda x_j(t) + (1-\lambda)\{\alpha F[x_j(t)] + I_j(t) - \beta \sum_{j \neq i} F[x_j(t)] + N(0, \sigma)\}$$

Equation 1: activation calculation of the context-activation model

The activation level of each item in the buffer, x_i , is determined by the items activation on the previous time step, self-recurrent excitation that each item recycles onto itself α , sensory input l , inhibition from the other active items β , and zero-mean Gaussian noise ξ with standard deviation σ . λ is the Euler integration constant that discretizes the differential equation.

Figure 2 illustrates the interplay between these competing forces in noiseless runs of the buffer when five pieces of data have been presented to the model for a fast rate of 100 iterations (top panel) and for a slower rate of 1500 iterations (bottom panel). The activation of each data rises as it is presented to the model and its bottom-up sensory input contributes to the activation. These activations are then dampened in the absence of bottom-up input as inhibition from the other items drive activation down. Self-recurrency can keep an item in the buffer in the absence of bottom-up input, but this ability is in proportion to the amount of competition from other items in the buffer. As can be seen, the fast presentation rate, in comparison to the slow rate, results in less competition from later items as the truncation of sensory input renders them less competitive. Importantly, this shift from recency to primacy with increasing presentation rate is a unique prediction made by this dynamic buffer and challenges other buffer models (Davelaar, et al., 2005).

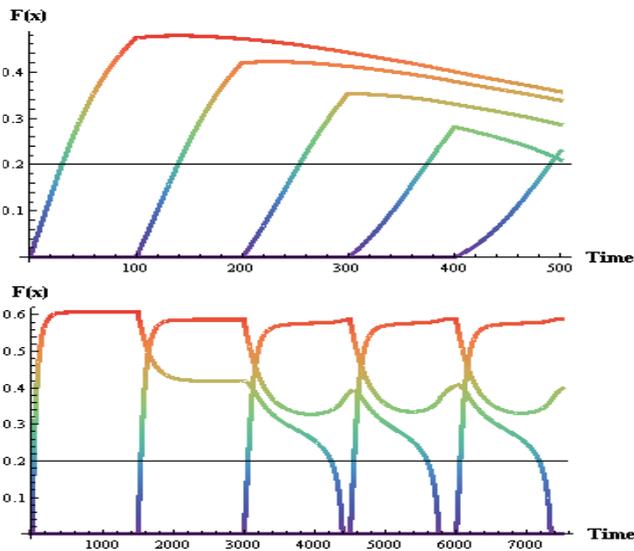


Figure 2: Activation trajectories for 5 sequentially received data at fast presentation rate (top) and slow presentation rate (bottom)

HyGene utilizes a representation from the multiple trace global matching models of MINERVA II (Hintzman, 1986, 1988) and the decision making model MINERVA-DM (Dougherty et al., 1999)¹. Separate episodic and semantic

¹ For a more thorough treatment of HyGene's computational architecture please see Thomas, Dougherty, Harbison, & Sprenger, (2008) or Dougherty, Thomas, & Lange (2010)

memory stores are present in the model. While semantic memory stores only individual prototypes of each disease, each experience the model acquires is represented in episodic LTM as a series of concatenated minivectors of 1s, 0s, & -1s where each minivector represents a hypothesis or data. That is, each trace is made up of one hypothesis and several pieces of data (in our case four). Retrieval is initiated when D_{obs} are matched against the data minivectors in LTM. This results in an activation level for each trace where a greater overlap in features present in the trace and in the D_{obs} results in greater activation. The weightings from the data acquisition buffer are used to weight the activations of each minivector in episodic memory at this point in retrieval. Therefore, the activation levels associated with each trace are directly influenced by the weightings for each data supplied by the dynamic working memory processes of the buffer.

Once these activation values have been obtained, only a subset of the episodic traces activated over a criterion are used for further processing in the model. From this subset of traces a probe is derived as a cue to semantic memory for the generation of hypotheses. This cue is matched against all known hypotheses in semantic memory. The activation values for each hypothesis serve as input into sampling via Luce's choice rule. Generation proceeds until a stopping rule is reached based on the total number of resamplings of previously generated hypotheses (i.e., retrieval failures).

We now present two experiments investigating separate consequences of hypothesis generation being extended over time. The first experiment examines how the mere serial position of a diagnostic datum influences the generation of the hypothesis it implies. Experiment two examines how processing time (i.e., presentation duration) per datum influences the contributions of the individual data in the generation process. The novel model of dynamic data acquisition and hypothesis generation discussed above is used to simulate the findings from both experiments. Critically, although many instantiations of a working memory buffer may predict the results from Experiment 1, the results from Experiment 2 provide support for our specific buffer instantiation, as borrowed from the context-activation model, underlying data acquisition in hypothesis generation tasks.

The Influence of Data Acquisition Dynamics on Hypothesis Generation

Experiment 1

The generalized order effect paradigm was developed by Anderson (1973) to examine the differential weighting of descriptive attributes presented in impression formation tasks. The procedure involved embedding a fixed list of information with a critical piece of information at various serial positions thereby allowing differences in the final rating to be uniquely attributable to the serial position of the critical data. The present experiment represents an

adaptation of this paradigm to a simulated medical diagnosis task to assess the impact of specific data serial positions on hypothesis generation.

Method

Participants Seventy-two participants participated in this experiment for course credit.

Design The design of Experiment 1 was a one-way between-subjects design with data order as the independent variable. The ecology for this experiment as defined by the conditional probabilities between the hypotheses and data is shown in Table 1. Each of the values appearing in this table represents the probability that the symptom will be present (e.g., fever) given a particular hypothesis whereas the complementary probability represents the probability of the symptom absence. As demonstrated in the table, the only diagnostic piece of data was D1 whereas the remaining cues, D2-D4, were non-diagnostic.

Table 1: Disease (Hypothesis) x Symptom (Data) ecology of Experiment 1.

		Symptoms			
		D1	D2	D3	D4
Diseases	H1: Metalytis	0.8	0.6	0.6	0.6
	H2: Zymosis	0.2	0.6	0.6	0.6
	H3: Gwaronia	0.2	0.6	0.6	0.6

Table 2 displays the four data orders. Each of these orders was identical (D2 → D3 → D4) except for the position of the D1 data within them.

Table 2: Data presentation orders.

	→ Presentation Position →			
	1	2	3	4
Order 1	D1	D2	D3	D4
Order 2	D2	D1	D3	D4
Order 3	D2	D3	D1	D4
Order 4	D2	D3	D4	D1

Procedure The procedure was comprised of two stages. The first stage was an exemplar training task in which a series of hypothetical pre-diagnosed patients was presented to the participant in order for them to learn the contingencies between the hypotheses and data through repeated experience. Each of these patients was represented by a diagnosis at the top of the screen (H1, H2, or H3) and a series of test results (i.e., symptoms) pertaining to the columns of D1, D2, D3, and D4. Over the course of the training phase the specific test results precisely respected the disease-symptom contingencies appearing in Table 1.

Following an arithmetic distraction task, the second stage of the procedure commenced. This was an elicitation phase in which we implemented our manipulation of data order and assessed hypothesis generation performance. The participants were then told that they were now going to see an individual patient’s symptoms and would then be asked to report the most likely diagnosis for the patient. The participant triggered the onset of the patient’s data stream at their readiness. Each datum of was presented individually for 1.5 seconds. The order in which the data were presented was determined by the order conditions as shown in Table 2. Following the presentation of the last datum the participant responded with the most likely disease.

Results

Empirical Nominal logistic regression was carried out on the generation data to examine the effect of data serial position on the generation of H1 (Metalytis), the disease with the greatest posterior probability given the data. There was a significant trend for H1 being reported as the most likely hypothesis as the serial position of the diagnostic data increased, $\chi^2(1) = 4.32, p < 0.05$.

Computational To simulate Experiment 1, the model’s episodic memory was endowed with the Hypothesis-Data contingencies described in Table 1. On each trial each piece of data was presented to the buffer for 1500 iterations (mapping onto the presentation duration of 1500 ms) and the order of the data was manipulated to match the data orders used in the experiment. 1000 iterations of the entire simulation were run for each condition². The model data was supplemented with a constant guessing parameter of 0.31 across all conditions.

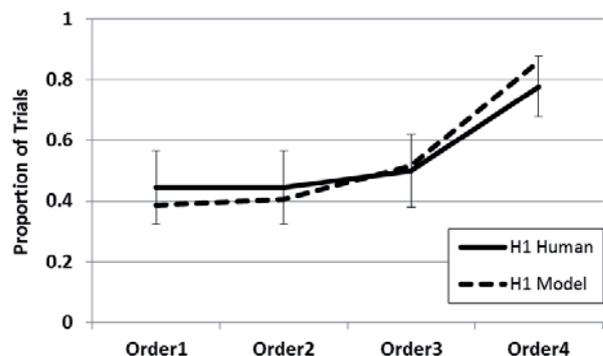


Figure 3: Human and Model results for Experiment 1 plotting the probability of reporting H1 as most likely across order conditions. Error bars represent standard errors.

As is demonstrated in Figure 3, the model is able to capture the empirical data quite well. This effect is directly attributable to the weights from the buffer being applied to the generation process.

² The parameters used for this simulation were the following. HyGene: L=0.85, Ac=0.075, Phi=4, KMAX=8 Buffer: Alpha=2.0, Beta=0.2, Lambda=0.98, Delta=1.0

Experiment 2

Method

Participants One hundred and twenty four participants participated in this experiment for course credit.

Design The design of Experiment 2 was a one-way between-subjects design with the presentation rate of the data as the independent variable. The ecology for this experiment appears in Table 3. The important aspect of this ecology is that the early data (D1 and D2) are diagnostic in favor of H1 whereas the later data (D4 and D5) favors H2.

Table 3: Disease (Hypothesis) x Symptom (Data) ecology of Experiment 2.

	D1	D2	D3	D4	D5
H1: Metalytis	0.8	0.7	0.5	0.3	0.3
H2: Zymosis	0.3	0.3	0.5	0.7	0.8

Procedure The procedure of this experiment was very similar to that of Experiment 1. Participants learned the hypothesis-data contingencies in an exemplar training phase prior to elicitation. However, in between these two phases of the experiment there was a learning test to discriminate participants that had learned the contingencies well from those that did not learn the contingencies. For this test the participants were provided with an individual piece of data and asked what the most likely hypothesis was. Their total learning score was the amount of correct responses in this task³.

In the elicitation phase the participants were provided with the data in the order in which they appear in Table 3, that is, consecutively from D1 to D5. Directly following the last piece of data the participant entered the disease they thought was most likely given the patient's symptoms. What varied between participants was the rate at which these data were presented. Half of the participants were presented the data at a fast rate (144 ms each) while the other half were presented the data at a slow rate (1504 ms each).

As displayed in Figure 2, the context-activation model predicts the fast presentation rate to lead to the earlier data residing more strongly in working memory following D5 whereas the model predicts the opposite for the slower presentation rate. Therefore we predicted that the fast presentation rate should lead to greater relative activations of early data thereby leading to greater generation of H1, whereas the opposite would be the case when the data are presented slowly leading to a preference for H2 and accordingly a lower rate of H1 relative to the fast condition.

Results

Empirical Although the rate of H1 selection was slightly higher in the fast presentation rate condition, this difference

³ Responses were counted automatically correct for responses to the D3 data as both hypotheses were equally likely.

did not reach significance, $z = 1.27$, $p = 0.102$. A further analysis was performed within groups of high learning and low learning participants based on their performance in the learning test. Those scoring higher than 60% were counted as high learners and those scoring lower were counted as low learners. Conditional analyses within each learning group revealed a marginal effect of presentation rate for the low learners, $z = 1.6$, $p = 0.054$ and no effect for the high learners, $z = 0.34$, $p = 0.367$. This result reflects the fact that the trend witnessed in the overall data was, somewhat counter-intuitively, due to those that did not learn the contingencies in the task as fully. We explain this effect below with our model.

Computational To simulate Experiment 2 the model was endowed with experience in the ecology displayed in Table 3. The manipulation of presentation rate was implemented in the model by varying the number of iterations the model was presented each piece of data. For instance, in the fast condition each piece of data received bottom-up input for 100 iterations whereas in the slow condition each piece of data received bottom up activation for 1500 iterations.

In line with the empirical result, however, we are not solely concerned with capturing differences in presentation rate, but we are additionally interested in capturing the difference that manifesting between the high and low learning groups. We posit this difference to be attributable to the role of working memory capacity (WMC). It is likely that high capacity participants were better able to learn the contingencies as each exemplar provided several bits of information for encoding. Furthermore, successful learning likely included some form of hypothesis testing carried out over successive exemplars which would be cognitively taxing, but beneficial to learning. Therefore we suggest that the high learning group possessed a greater proportion of high capacity participants.

In the present analysis, we ask if differences in a parameter governing the emergent capacity of the buffer could explain the presence of a presentation rate effect amongst low learners and its amelioration amongst high learners. As the beta parameter governs the strength of the global inhibition that is applied to each item this parameter can be used to impose capacity constraints (Davelaar, 2007). As beta is increased, competition between items is increased and fewer items will cohabit the buffer. We manipulated beta at two levels to capture low learning/low capacity (beta= 0.1) and high learning/capacity (beta=0.05) and used presentation rates of 100 iterations (fast rate) and 1500 iterations (slow rate)⁴. This resulted in averaged summed activations in the buffer of 2.22 in the slow rate and 2.33 in the fast rate under beta=0.05 and values of 1.96 in the slow rate and 1.74 in the fast rate under beta=0.1. Therefore, more activation was present in working memory on average when beta=0.05 relative to beta=0.1. This entire simulation was run for 700 model runs of each condition.

⁴ All other model parameters were the same as those used for Experiment 1.

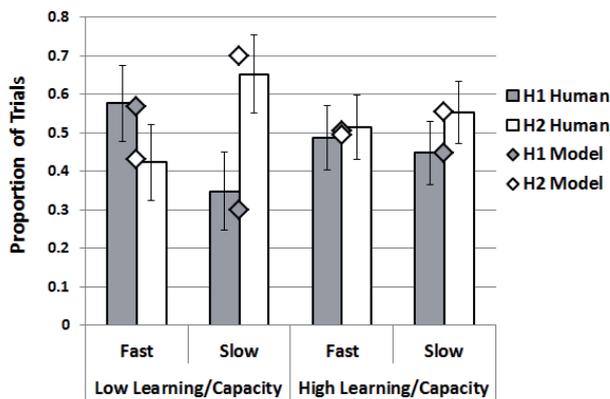


Figure 5: Human and Model results for Experiment 2 plotting the probability of generating H1 and H2 as most likely by presentation rate and learning/capacity groups. Error bars represent standard errors.

As demonstrated in Figure 5, the model is able to capture the patterns in the data occurring with differences in presentation rate and differences in learning between participants.

Discussion

We presented and tested an extension of the HyGene model. The model was endowed with a dynamic working memory buffer and adequately captured a recency bias in generation (Experiment 1). In addition, the sensitivity of the dynamic buffer to presentation rate was shown to influence hypothesis generation (Experiment 2). Moreover, individual differences in learning or WMC interacted with the balance of incorporating primacy and recency items in the decision. Moreover, the ability of our model to capture the results from both experiments lends credence to our specific buffer implementation.

The present work demonstrates the utility of understanding working memory dynamics during *data acquisition* (cf. Mehlhorn et al., 2011) and suggests that the activations of individual pieces of data in working memory govern their individual contributions to the hypothesis generation process. The model presented here will be extended in future work such that the activations of hypotheses themselves will be subject to the competitive buffer dynamics demonstrated here. This model will address the hypothesis maintenance and updating components of temporally dynamics hypothesis generation and utilization following retrieval from long-term memory.

Acknowledgments

This research was supported by a grant from the US National Science Foundation (#SES-1024650) awarded to Rick Thomas, Nicholas Lange, and Eddy Davelaar.

References

- Anderson, N. H. (1973). Serial position curves in impression formation. *Journal of Experimental Psychology*, *97*(1), 8-12.
- Davelaar (2007). Sequential retrieval and inhibition of parallel (re)activated representations: A neurocomputational comparison of competitive cueing and resampling models. *Adaptive Behavior* (*15*)1, 51-71.
- Davelaar, E. J., Goshen-Gottstein, Y., Ashkenazi, A., Haarmann, H. J., & Usher, M. (2005). The demise of short term memory revisited: Empirical and computational investigations of recency effects. *Psychological Review*, *112*(1), 3-42.
- Dougherty, M.R.P., Gettys, C. F., & Ogden, E. E. (1999). A memory processes model for judgments of likelihood. *Psychological Review*, *106*(1), 180-209.
- Dougherty, M. R. P., & Hunter, J. E. (2003)a. Probability judgment and subadditivity: The role of WMC and constraining retrieval. *Memory & Cognition*, *31*(6), 968-982.
- Dougherty, M. R. P., & Hunter, J. E. (2003)b. Hypothesis generation, probability judgment, and working memory capacity. *Acta Psychologica*, *113*(3), 263-282.
- Dougherty, M. R., Thomas, R. P., & Lange, N. (2010). Toward an integrative theory of hypothesis generation, probability judgment, and information search. In B. H. Ross (Ed.), *The Psychology of Learning and Motivation* (Vol. 52). San Diego: Elsevier.
- Hintzman, D. L. (1986). "Schema Abstraction" in a multiple-trace memory model. *Psychological Review*, *93*, 411-428.
- Hintzman, D. L. (1988). Judgments of frequency and recognition memory in a multiple-trace memory model. *Psychological Review*, *95*, 528-551.
- Mehlhorn, K., Taatgen, N. A., Lebiere, C., & Krams, J. F. (2011). Memory activation and the availability of explanations in sequential diagnostic reasoning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *37*(6), 1391-1411.
- Sprenger, A. & Dougherty M.R. (2011). Generating and evaluating options for decision making: The impact of sequential presented evidence. *Journal of Experimental Psychology: LMC* doi: 10.1037/a0026036
- Sprenger, A. & Dougherty, M. R. (2006). Differences between probability and frequency judgments: The role of individual differences in working memory capacity. *Organizational Behavior and Human Decision Processes*, *99*(2), 202-211.
- Sprenger, A. M., Dougherty, M. R., Atkins, S. M., Franco-Watkins, A. M., Thomas, R. P., Lange, N., & Abbs, B. (2011). Implications of cognitive load for hypothesis generation and probability judgment. *Frontiers in Psychology-Cognitive Science*, *2*, 1-15
- Thomas, R. P, Dougherty, M. R., Sprenger, A. M., & Harbison, J. I. (2008). Diagnostic hypothesis generation and human judgment. *Psychological Review*, *115*(1), 155-185.

A cognitive architecture-based model of graph comprehension

David Peebles (d.peebles@hud.ac.uk)

Department of Behavioural and Social Sciences,
University of Huddersfield,
Queensgate, Huddersfield, HD1 3DH, UK

Abstract

I present a model of expert comprehension performance for 2×2 “interaction” graphs typically used to present data from two-way factorial research designs. Developed using the ACT-R cognitive architecture, the model simulates the cognitive and perceptual operations involved in interpreting interaction graphs and provides a detailed characterisation of the information extracted from the diagram, the prior knowledge required to interpret interaction graphs, and the knowledge generated during the comprehension process. The model produces a scan path of attention fixations and a symbolic description of the interpretation which can be compared to human eye movement and verbal protocol data respectively, provides an account of the strategic processes that control comprehension, and makes explicit what underlies the differences between expert and novice performance.

Keywords: Graph comprehension, ACT-R

Introduction

Working with graphs is a complex skill that requires specific knowledge of the representational system being used together with a set of procedures to map spatially represented information in the graph with a set of propositions that specify quantitative and qualitative relationships between the entities represented. Providing a detailed account of this skill therefore requires one to specify a number of core assumptions including: what and how information is encoded in the diagram, what and when information is obtained from the diagram by the user during a task, what and how prior graph knowledge is stored and utilised, and what new knowledge is created during the process. In addition, one must also specify the strategies people employ to carry out different tasks and how much these strategies use information in the diagram and in stored internal representations.

There have been several attempts to provide detailed process models of different aspects of graph use. Models are constructed from sets of perceptual and cognitive operators (e.g., encode the value of an indicator, make a spatial comparison between indicators (Gillan, 1994), compare two digits in working memory, or make a saccade (Lohse, 1993)), obtained either from task or verbal protocol analyses. Lohse (1993) and Gillan (1994) have produced models of question answering with several different graph types (including line graphs, bar charts and scatter plots) by constructing sequences of operators (each of which has an associated execution time) to generate predicted scan paths across the graph and total task completion times which can be compared to human data.

Other researchers have procedurally analysed graph use for different purposes. For example, Casner (1991) identified a set of perceptual and cognitive operators to construct models of several graph-based tasks which informed an automated system that generated graphical representations most

suitable to the tasks commonly undertaken with them. A similar method was adopted by Tabachneck-Schijf, Leonardo, and Simon (1997) in their analysis of an economics expert’s construction of a graph while explaining the principle of supply and demand which they then used to develop a computational model incorporating both diagrammatic and propositional representations.

More recently, the cognitive modelling of reasoning with information displays has been advanced by the development of *cognitive architectures*; computational theories of the large-scale structure of the mind providing accounts of how cognition is controlled and how knowledge is encoded, stored, retrieved and utilised (e.g., ACT-R (Anderson, 2007), EPIC (Meyer & Kieras, 1997), and Soar (Laird, Newell, & Rosenbloom, 1987)).

The first two of these architectures incorporate theories of visual processing and motor control which allows modellers to produce more detailed accounts of the information obtained from the display during the task. For example Peebles and Cheng (2003) used ACT-R to produce a computational model of question answering using two different types of line graph. Their model generated saccades and fixations as it answered each question which, together with task completion times, were compared to human data. In addition, the model was able to account for human scan paths in terms of the varying demands on memory imposed by different questions.

The Peebles and Cheng study, as did those by Lohse (1993) and Gillan (1994), investigated question answering in which participants were given items of information and were required to produce associated information using different processes, including identification (e.g., “In 1997, what was the value of gas?” (Peebles & Cheng, 2003)), comparison (e.g., “In 1977 did tin cost less than sulphur?” (Lohse, 1993)), and arithmetic computation (e.g., “What is the sum of A, B, and C?” (Gillan, 1994)).

While these are important tasks, particularly for investigating sequences of elementary processes, it could be argued that they do not necessarily reflect how many people normally work with graphs and that they do not address the important prior comprehension stage where labels and graphical features are encoded, associated, and interpreted (Carpenter & Shah, 1998).

Comprehension requires knowledge of the conventions used in the graph to represent data and other facts such as how labels are to be interpreted based on their location. The output of the process is assumed to be a set of knowledge structures that represent the variables and graphical features together with structures that encode knowledge about the quantitative

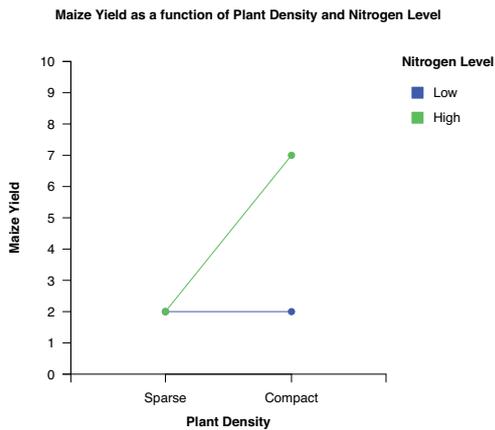


Figure 1: One of the eight line graphs used in the expert study (Peebles & Ali, in preparation).

or qualitative relationships between the variables depicted.

A prime example of a scenario where people encounter a graph with the sole aim of comprehending the relationships between variables (as opposed to identifying trends or individual values for example) is the analysis of data from factorial experiments. The simplest form of factorial design is the *two-way factorial design*, containing two factors, each with two levels, and one DV. Statistical analysis of these designs typically results in a 2×2 matrix of mean values of the DV corresponding to the pairwise combination of the two levels of each IV. Interpreting the results of even these simplest of designs accurately and thoroughly is often not straightforward however, but requires a significant amount of conceptual understanding—for example the concepts of simple, main, and interaction effects. As with most other statistical analyses however, interpretation can be eased considerably by representing the data in diagrammatic form.

Data from two-way factorial designs are most often presented as either line or bar graphs—variously called *interaction* or *ANOVA* graphs. An examples line graph is shown in Figure 1. Because the data come from pair-wise combinations of the IV levels, the rules for interpreting interaction graphs are quite specific however and sufficiently different from other more frequently encountered line graphs that simply applying general interpretive rules will not prove particularly helpful (other than for obtaining the DV values of specific conditions etc.). The key elements of knowledge to be obtained from interaction graphs are the simple, main and interaction effects of the IVs and these have to be identified in specific features of the graph.

In a series of studies, Peebles and Ali have observed and recorded novices (undergraduate psychology students) and experts (cognitive science professors and postgraduate researchers) interpreting interaction graphs like the one in Figure 1 (Peebles & Ali, 2009; Ali & Peebles, in press; Peebles

& Ali, in preparation). These studies have shown that without knowledge of the appropriate interpretive rules, novices' interpretations are often limited to qualitative descriptions of differences between conditions and can be skewed by the different Gestalt principles of perceptual organisation (Wertheimer, 1938) operating in the graph. In contrast, expert users are able to employ their knowledge of which graphical features represent which effects to identify relationships between variables much more rapidly and accurately with no prior knowledge of the domain variables being represented in the graph.

The purpose of the research reported here is to develop a computational model of graph comprehension that specifies the processes underlying both expert and novice behaviour with sufficient detail and comprehensiveness to satisfy all of the criteria outlined at the beginning of this paper. Specifically, the model aims to provide a precise account of the minimum information required to interpret interaction graphs appropriately together with a hypothesis as to the nature of the processes involved in representing and interpreting that information. The model is developed within the ACT-R cognitive architecture and therefore embodies assumptions about the nature of the mental representations and the computations that form the strategies used to generate new representations. Finally, the model provides an explanation for the differences between expert and novice interpretations.

A model of graph comprehension

Space limitations preclude a detailed description of ACT-R here. However a comprehensive account of the cognitive architecture can be found in Anderson (2007). In summary, ACT-R consists of a set of modules that acquire information from the environment, process information, and execute motor actions to achieve goals. ACT-R has memory stores for declarative and procedural knowledge. The former consists of a network of knowledge chunks while the latter is a set of production rules. Cognition proceeds via a pattern matching process that attempts to find production rules with conditions that match the current state of the system and tasks are performed through the successive actions of production rules.

ACT-R also incorporates a subsymbolic level of computations that govern memory retrieval and production rule selection and which allow models to account for widely observed recency and frequency effects on retrieval and forgetting. Subsymbolic computations also underlie ACT-R's different learning mechanisms.

For tasks involving displays and other devices, task environments can be defined to be acted upon by the model. The graphs used in this study are defined as sets of visual objects (lines, circles, rectangles, and text) with certain features (size, colour) at specific x-y coordinates on a 2D window.

The graph comprehension model is based on verbal protocol data from novice and expert users (Peebles & Ali, 2009; Ali & Peebles, in press; Peebles & Ali, in preparation). In these studies, verbal statements recorded during the compre-

hension task were coded and categorised in terms of their functional role and content (e.g., “an association between a level and its identifier”; “a comparison between the two legend variable levels for one of the levels of the x axis variable”) to produce a set of common interpretive operations.

The verbal protocols indicate that comprehension is typically carried out in two main phases: (a) a variable identification stage followed by (b) a pattern recognition and description stage. The protocols also reveal that experts and a large proportion of novices rarely report specific DV values, but typically produce qualitative descriptions of the differences between conditions.

In the first stage, the three variables are identified, categorised as dependent or independent according to location, and the latter associated with their levels, which in turn are associated with identifiers (left or right position for the x axis variable and colour for the legend variable).

In the second stage, distances between plot points are observed and compared, with the results being used to probe long-term declarative memory for interpretive knowledge. If this is successful, the retrieved knowledge is used to provide an interpretation. If there is no interpretation available however, the model will simply describe the identification or comparison process being carried out. Interpretive operations are carried out until either a full interpretation is produced or until no other operations are available or identified.

Representing and encoding information in the graph

The key information that the model must encode and utilise from the graph representation is a set of four x-y coordinate points and the spatial distances between them. Although the model processes symbolic representations, it assumes that spatial information is initially encoded quantitatively and subsequently categorised into qualitative descriptions. The perceptual processes by which the information is obtained or represented are not specified in detail, although it is assumed that it is via a set of prior *elementary perceptual tasks* (Cleveland & McGill, 1984). Cleveland and McGill (1984) identified ten such tasks (e.g., length, direction, area, and position on a common scale) as the “perceptual building blocks” of graph comprehension that encode quantitative information from graphical elements.

At least two such elementary perceptual tasks are assumed to be required for these graphs. The first is *position on a common scale* and this is the primary comparison that takes place. It is assumed that readers initially encode the spatial distance between plot points into a quantitative representation (the proportion, p , of the distance to the overall length of the y axis) and then categorise this ratio according to size. For this model six categories were assumed: “no” ($p = 0$), “very small” ($0 < p < 0.2$), “small” ($0.2 = < p < 0.4$), “moderate” ($0.4 = < p < 0.6$), “large” ($0.6 = < p < 0.8$), and “very large” ($0.8 = < p < 1.0$). Although it is an assumption of the model that distances are categorised in this way, the exact processes by which these final categories are produced are not specified in detail.

The second process that is assumed readers can perform is to compare the magnitude of two distances and produce a symbolic description of the difference. The elements formed for this comparison are assumed to be the result of Gestalt processes of perceptual organisation (Ali & Peebles, in press; Kosslyn, 1989; Pinker, 1990) which allow users to group objects by colour or proximity. This comparison also allows users to perceive and compare the directions of the two differences (i.e., the relative sizes of the variables’ level values). These relative values produce the various patterns such as crossed, parallel and diverging lines which are recognised and interpreted by expert users.

Prior graph knowledge

Two forms of declarative knowledge are used during the task: prior knowledge relating to how the graph represents information and knowledge of the variables and their relationships generated during the comprehension process itself.

There are three core items of knowledge required to interpret interaction graphs. Two are common to many Cartesian graphs and concern (a) the typical allocation of the dependent and independent variables to the graph axes and legend and (b) the principle that the distance between two graphical elements is directly related to the magnitude of the relationship between the conceptual entities that the elements represent.

The third set of facts required are specific to the graph type and concern the graphical and spatial indicators of the three key important interpretive facts; simple effects, main effects, and interactions. The three indicators are (a) the distance between two plot points which indicates the size of the *simple effect* of the level jointly represented by those points, (b) differences in the y-axis location of the midpoints between two pairs of plot points which indicate the size of the *main effect* of the variable, and (c) differences in the inter-point distances between levels, combined with information about their point ordering, which indicates the size, and type of any interactions that may exist.

This knowledge is represented as symbolic structures in the model’s long term memory and is currently the minimum required to indicate that the interpretative process has succeeded. It is possible however to add further causal knowledge relating to the various effects to allow the model to provide more detailed explanations of the relationships identified.

Generated knowledge

Several declarative knowledge structures are also generated during comprehension. The first is a set of related chunks that represent each variable, the levels associated with it, and the identifiers of each level. Three other knowledge structures are generated to accumulate and associate items of graph and interpretive information during a specific sub-task. In the expert model all knowledge retrieval requests will succeed, resulting in knowledge structures that associate qualitative descriptions of differences and their interpretation. These structures could then be used to produce verbal explanations.

For example one structure records the processing of an individual level which could produce the explanation: “The difference between the two values for high plant density is very large so there’s a very large simple effect of high plant density” while another records the information accumulated when comparing the average values of two levels of one variable (e.g., “There is a large difference between the fasting levels; high fasting generally resulted in greater glucose uptake than low fasting, which indicates a large main effect of fasting”). The third stores the results of comparing the lengths and point ordering of two levels (e.g., “Although the effect size of the cement type levels is the same, the direction of their effects is different so that means there is an interaction between the two independent variables”).

Finally, a representation is produced when a simple comparison between points is made which does not associate an interpretation (e.g., “When the nitrogen level is high, maize yield is much greater for compact plants than for sparse plants”).

The comprehension process

In the Appendix is an output trace produced by the model as it carries out the comprehension task using the graph in Figure 1, with each line in the trace representing one or more steps in the process (variable names have been shortened to allow the lines to fit the format of this paper). The text in square brackets is information currently being processed that has either been obtained from the graph or retrieved from declarative memory¹.

In the trace, numbers in square brackets represent the perceptual difference between two objects on the screen. These are subsequently translated into qualitative size judgements according to the categories described above. Other text in the output is simply to indicate other events (e.g., goal setting or memory retrieval failures) or to clarify what a particular knowledge element represents.

As previously intimated, the model assumes that comprehension proceeds after an initial phase of variable identification, a process usually initiated by reading the title. Currently when the model reads the title the three words that name variables are identified by retrieving previously defined word category information from declarative memory. This mechanism is undoubtedly simplistic and currently substitutes for a more complex knowledge retrieval process that is assumed to take place.

The model then seeks items of text at the left, right and lower regions of the display. When each variable label is identified, the model identifies it as a particular type according to its location and then, associates the independent variables with their level labels by identifying nearby text. The model also associates each of the four levels with its physical attribute; left, right, blue and green and uses these labels when processing the graph. This is consistent with verbal

¹A video of the model interpreting all eight graphs from the expert study (Peebles & Ali, in preparation) can be viewed at <http://youtu.be/z2kAwr0rjIM>

protocol and eye movement data from our studies showing that graph readers often produce an interpretation and then must re-read the appropriate label in order to identify which particular level is being processed.

When the three variables have been processed, the model then attends to the pattern produced by the four coordinate points in the plot region and then selects a particular feature or pair of features to process. The probability of selecting a particular feature to process may depend on a number of factors, including visual salience and pattern familiarity. For example, a large difference between objects, or parallel or crossing lines may draw the user’s attention and lead them to attempt to interpret the feature first. Although it is possible to incorporate these processes for the model to select features in any order, for simplicity, the current model selects features in the order: simple, followed by main, and finally interaction effects.

These three effects are identified by different indicators in the graph. The size of the simple effect of a level is indicated by the distance between the level’s two plot points while the main effect of a variable is indicated by the difference in the y-axis location of the midpoints between the variable’s two pairs of plot points. Finally, the nature and size of interaction effects are indicated by differences in the inter-point distances between levels, combined with information about their point ordering.

The model represents the interpretation process by a set of production rules for each indicator type. When the appropriate condition occurs (i.e., the model is directing attention to the plot region), individual production rules fire to draw attention to specific indicators. The indicator (a spatial distance, difference or order comparison), is extracted from the pattern and (together with information about what the indicator is) used to probe declarative memory for an interpretation consisting of the name and size of the effect. For example on line 29 of the trace the model identifies that there is no difference between the plot points on the left of the display and then retrieves the knowledge that this indicates that there is no simple effect of sparse plant density (these labels being obtained by seeking the text below the points being observed).

For each indicator, if the memory retrieval attempt fails, the model simply describes the difference being attended to. This is demonstrated in lines 37 and 38 of the trace which compare the levels of the legend variable for each of the x axis variable levels and which correspond to the statement “when plant density is sparse, low and high nitrogen levels are the same but when plant density is compact, the high nitrogen level is greater than the low nitrogen level”. This form of statement is very common in novice graph users.

Once a recognition production rule fires to initiate the process, a chain of subsequent productions is triggered which obtains further information from the graph and declarative memory until an interpretation is produced. The current production set is sufficient to process any 2×2 data set of three variables to produce an appropriate interpretation similar to

the trace in the Appendix.

Discussion

Comprehending and reasoning with graphs requires a wide range of perceptual and cognitive operations sequenced together in various combinations to perform specific tasks. The type and sequence of operators involved in a task may differ depending on a number of factors, including the graph or domain knowledge of the user, the type of graph being used, or individual cognitive factors such as working memory capacity (which may determine the relative frequency of memory retrieval requests and saccades to graph labels etc.).

Graph comprehension is an important area to study therefore because it provides an opportunity to investigate how environmental and internal factors interact to produce behaviour. In addition, graph-based tasks can be analysed using behavioural measures such as eye movements and concurrent verbal protocols to provide insights into what and when information is being processed during the course of the activity.

Computational modelling is a valuable tool for developing and testing hypotheses about the representations and mechanisms necessary for cognitive tasks as it provides a formalism for characterising them, requires one to be explicit about the boundaries of one's model in terms of which processes are being defined precisely and which are not, and allows one to explore the consequences of particular assumptions (McClelland, 2009).

Developing models within a cognitive architecture such as ACT-R provides the additional benefit of allowing the model to incorporate a large number of assumptions regarding issues such as knowledge representation, cognitive control, visual attention, learning and forgetting etc., all of which are supported by previous empirical research. In addition, ACT-R's vision module includes mechanisms that allow models to simulate certain Gestalt principles of perceptual organisation, which are regarded as playing a crucial role in the visual processing of graphical representations (Kosslyn, 1989; Pinker, 1990). Specifically, the comprehension model associates variables and their levels, and levels with their colour identifiers using mechanisms that are functionally equivalent to the Gestalt laws of *proximity* and *similarity* respectively.

The model described above represents an initial attempt to specify at a detailed algorithmic level the representations, cognitive processes, and strategies involved in comprehending interaction graphs. It provides a precise account of the graph knowledge required and the spatial information necessary to interpret the graph accurately and specifies a control structure that determines the flow of information during the task to generate a set of knowledge representations, saccades and fixations over the graph, and a sequence of output statements which are largely consistent in terms of order, function and content with verbal protocols produced by expert users.

The assumptions of the model imply that to interpret interaction graphs accurately, novices must acquire three forms of graph-specific knowledge: an understanding of what effects

the different distances and spatial differences in the graph indicate, the relationship between distance and effect size, and how the various combinations of distance differences and point orders can be interpreted in terms of the interactions between the IVs. The model provides a precise specification of the relatively small amount of knowledge required and a clear demonstration of its sufficiency to interpret the graphs.

The current model can be considered a first approximation to a more detailed model that incorporates additional factors to broaden the scope of behaviour accounted for. Previous studies have shown that comprehension performance varies quite widely, even between experienced users (Peebles & Ali, 2009; Ali & Peebles, in press; Peebles & Ali, in preparation). For example, the order in which effects were identified varied, often as a result of the relative visual salience of the graphical features being displayed (e.g., very large main effects were often identified rapidly). Also, explicitly identifying simple effects was uncommon and other effects were sometimes overlooked by experienced users.

This variation in performance is no doubt due to a number of factors including the different effects of visual salience and Gestalt principles of perceptual organisation operating (Ali & Peebles, in press), and varying levels of graph knowledge and working memory capacity etc. In addition, previous studies compared expert and novice performance on both bar and line graph formats and showed that the interpretations of all users (but novices in particular) were affected by the format used. Specifically, line graphs users are influenced to attend to the legend variable while bar graph users attend to the two IVs more equally (Peebles & Ali, 2009; Ali & Peebles, in press; Peebles & Ali, in preparation). Broadening the scope of the model further, other factors such as domain knowledge and the number of variable levels (Shah & Freedman, 2011) should also be addressed.

The current model provides a solid basis from which to explore hypotheses concerning the mechanisms underlying this broader range of behaviour. These hypotheses will take the form of enhanced or reduced declarative graph or domain knowledge, additional recognition productions, and mechanisms to represent visual salience. A more comprehensive model must also bring ACT-R's subsymbolic mechanisms that govern memory retention, retrieval, and learning processes into play as these no doubt have a significant effect on strategy choice and eye movement patterns (Peebles & Cheng, 2003).

Finally, the current model does not attempt to provide a detailed account of the perceptual processes by which spatial information is encoded or represented during the execution of elementary perceptual tasks. There are currently several attempts to develop mechanisms for spatial representation and processing within cognitive architectures—including ACT-R—however (a number of which are presented in (Gunzelmann, 2011)) and it may be possible for the current functions to be replaced in a future model by ones more conforming with theory and empirical evidence.

Beyond the goal of developing the model to account for the full range of observed behaviour with an increasing number of interaction graph formats, lies the larger aim of constructing a model of comprehension for a broader class of graphs. As discussed earlier, interaction graphs embody a specific set of interpretive rules that are not shared by other graphs. The current model clearly identifies and characterises these rules and distinguishes them from the knowledge and procedures that can be applied to other graphs. It is hoped that in so doing, the model will simplify the task of identifying graph-specific operators and form a basis upon which to develop and explore a range of graph comprehension models for other graphical formats. As it stands however, the model provides a valuable demonstration that the assumptions it currently embodies are sufficient to produce an expert interpretation of the relationships depicted in 2×2 interaction graphs.

Appendix: Model output for the graph in Figure 1

1 seek text at top of display...
 2 [m-yield] = [variable]
 3 [as] [a] [function] [of] [p-density] = [variable]
 4 [and] [n-level] = [variable]
 5 seek text at far right of display...
 6 [n-level] at [far right] = [independent] variable
 7 look to nearest text...
 8 [low] = level of [n-level]
 9 [high] = level of [n-level]
 10 seek objects in plot region...
 11 [blue] [line]
 12 no memory for [blue] look to legend
 13 [blue] [rectangle]. looking for nearest text... [blue] = [low]
 14 [green] [rectangle]. looking for nearest text... [green] = [high]
 15 seek text at far left of display...
 16 [m-yield] at [far left] = [dependent] variable
 17 seek text at bottom of display...
 18 [p-density] at [bottom] = [independent] variable
 19 look to nearest text...
 20 [compact] = level of [p-density]. [compact] = [right]
 21 [sparse] = level of [p-density]. [sparse] = [left]
 22 identify legend levels...
 23 [0.0] diff [blue] so [no] [simple] effect [low] [n-level]
 24 [0.5] diff [green] so [moderate] [simple] effect [high] [n-level]
 25 compare [blue] & [green] levels...
 26 [small] diff. [high] [n-level] > [low] [n-level]
 27 [small] [main] effect [n-level]
 28 identify x axis levels...
 29 [0.0] diff [left] so [no] [simple] effect [sparse] [p-density]
 30 [0.5] diff [right] so [moderate] [simple] effect [compact] [p-density]
 31 compare [left] & [right] levels...
 32 [small] diff. [compact] [p-density] > [sparse] [p-density]
 33 [small] [main] effect [p-density]
 34 compare left and right patterns...
 35 [0.5] diff in distance between points. [right] bigger
 36 [moderate] diff & [same] point order so [moderate] [interaction]
 37 for [sparse] [p-density] [low] [n-level] = [high] [n-level]
 38 for [compact] [p-density] [high] [n-level] > [low] [n-level]

References

- Ali, N., & Peebles, D. (in press). The effect of Gestalt laws of perceptual organisation on the comprehension of three-variable bar and line graphs. *Human Factors*.
- Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York, NY: Oxford University Press.
- Carpenter, P. A., & Shah, P. (1998). A model of the perceptual and conceptual processes in graph comprehension. *Journal of Experimental Psychology: Applied*, 4(2), 75–100.
- Casner, S. M. (1991). A task-analytic approach to the automated design of graphic presentations. *ACM Transactions on Graphics*, 10, 111–151.
- Cleveland, W. S., & McGill, R. (1984). Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association*, 79, 531–554.
- Gillan, D. J. (1994). A componential model of human interaction with graphs: 1. linear regression modelling. *Human Factors*, 36(3), 419–440.
- Gunzelmann, G. (Ed.). (2011). *Modeling spatial cognition [special issue]* (Vol. 3) (No. 4). Topics in Cognitive Science.
- Kosslyn, S. M. (1989). Understanding charts and graphs. *Applied Cognitive Psychology*, 3, 185–226.
- Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). SOAR: An architecture for general intelligence. *Artificial Intelligence*, 33, 1–64.
- Lohse, G. L. (1993). A cognitive model for understanding graphical perception. *Human-Computer Interaction*, 8, 353–388.
- McClelland, J. L. (2009). The place of modeling in cognitive science. *Topics in Cognitive Science*, 1(1), 11–38.
- Meyer, D. E., & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance 1. basic mechanisms. *Psychological Review*, 104, 3–65.
- Peebles, D., & Ali, N. (2009). Differences in comprehensibility between three-variable bar and line graphs. In *Proceedings of the thirty-first annual conference of the cognitive science society* (pp. 2938–2943). Mahwah, NJ: Lawrence Erlbaum Associates.
- Peebles, D., & Ali, N. (in preparation). *Comparing novice and expert comprehension of three-variable bar and line graphs*.
- Peebles, D., & Cheng, P. C.-H. (2003). Modeling the effect of task and graphical representation on response latency in a graph reading task. *Human Factors*, 45, 28–45.
- Pinker, S. (1990). A theory of graph comprehension. In R. Freedle (Ed.), *Artificial intelligence and the future of testing* (pp. 73–126). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Shah, P., & Freedman, E. G. (2011). Bar and line graph comprehension: An interaction between top-down and bottom-up processes. *Topics in Cognitive Science*, 3(3), 560–578.
- Tabachneck-Schijf, H. J. M., Leonardo, A. M., & Simon, H. A. (1997). CaMeRa: A computational model of multiple representations. *Cognitive Science*, 21, 305–350.
- Wertheimer, M. (1938). Laws of organization in perceptual forms. In W. D. Ellis (Ed.), *A source book of Gestalt psychology*. London: Routledge & Kegan Paul.

Cognitive modelling of early music reading skill acquisition for piano

Bruno Emond (Bruno.emond@nrc-cnrc.gc.ca)
National Research Council Canada
1200 Montreal Rd., M50, Ottawa, ON, Canada K1A 0R6

Gilles Comeau (gcomeau@uottawa.ca)
School of Music, University of Ottawa
50 University, Ottawa ON Canada K1N 6N5

Abstract

In the classical music tradition, knowing how to read music is an essential skill and is seen as a fundamental component to develop when learning to play the piano. This research's focus is to study the possible impact of the different teaching approach on the acquisition of initial reading skills. By using cognitive modeling, we are hoping to observe through computer simulation the problem solving and decision-making tasks involved in decoding a simple musical score. The paper introduces the Middle-C and Intervallic methods followed by a description of an ACT-R cognitive model and simulation results upon learning with each of the piano methods.

Keywords: Music reading, piano methods, ACT-R.

Introduction

In the classical music tradition, knowing how to read music is an essential skill and is seen as a fundamental component to develop when learning to play the piano (Galyen, 2005; Sloboda, 2005). However, learning to read musical notation is a long and arduous undertaking (Anderson, 1981; Hahn, 1985) and, despite the value we attribute to it, it is not always successful. In North America and in Europe, piano book tutors are at the centre of a beginner student's learning environment as piano teachers often rely on these books to provide the whole foundation of a pianist's musical education and much of the initial training on reading musical notation (Stewart, Henson, Kampe, Walch, Turner & Frith, 2003; McPherson & Gabrielsson, 2002). However, while having music reading as a common objective, the book tutors have introduced fundamentally different approaches such as the Middle-C, Intervallic or Multi-key approach; and more recently the Eclectic or Modified Multiple Key approach, which has supplanted the original Multi-key (Lomax, 1990). Surprisingly, despite the fact that the main focus of the piano tutors is the development of music reading skills, little is known about how this is done. Piano pedagogy textbooks provide long list of advantages and disadvantages for each of the different teaching approach (Uszler, Gordon & Smith, 2000), however it is all based on intuition and on teachers experience and it has no experimental basis to support the analysis, or formal model of its development. Little scientific information is available to evaluate the real impact of each reading systems, to establish their efficacy and efficiency.

It is well recognized that there is a lack of cognitive models to explain how music reading is acquired. Hodges, the author of the *Handbook of Music Psychology* (1996) and author of a chapter on music reading in the *Handbook of Research in Music Teaching and Learning* (1992) wrote that "in music there is no theory devoted specifically to an explanation of music reading; thus, the bulk of the research appears to be devoid of a theoretical underpinning" (1992, p. 469). Sixteen years later, he confirmed that the situation was still the same (Lemay, 2008). The few theoretical models that have been proposed over the years are either still in an embryonic stages or entirely speculative and devoid of an experimental basis (Udtaisuk, 2005). The most well-known cognitive model of music sight-reading was published by Wolf in 1976, and it was developed entirely based on interviews with four pianists (Wolf, 1976). It explains sight-reading as a problem-solving activity of pattern recognition, but no quantitative investigations were undertaken to refine and give legitimacy to the model. Fifteen years ago, Waters, Townsend and Underwood (1998) realized a series of laboratory experimentation to observe how pattern recognition' skills could play an important role in expertise musical sight reading and they have shown that in the pattern-recognition task, immediate recall of presented material correlate strongly with good sight-reading skills. Their study confirmed various experimentations conducted previously by Sloboda (1978, 1985) to show the importance of pattern recognition in various tasks related to music reading. However, while pattern recognition seemed to be a promising avenue to help our understanding of music reading skills, Madell and Hébert (2008) deplore the fact that more recent trends in music reading research has been to experiment with the intricacy of eye tracking technology without a focus on pattern recognition (Kinsgler and Carpenter, 1995). In addition, music reading studies deals with musicians who already know how to read music and have often reach the level of expertise. These models do not always shed lights on the skills required by a novice just being introduced to music notation. Without a solid model of music reading acquisition, it is not surprising that piano teaching material have come to propose very different approaches to music reading.

Piano playing is an elaborate skill that requires the coordination of many cognitive resources and subtle body movements. As such, expert piano playing performance has been the subject of many investigations (Hallam, Cross &

Thaut, 2009; Altenmüller, Wiesendanger & Kesselring, 2006; Parncutt & McPherson, 2002). However, the effect of pedagogical methods on novice performance and learning has not received the same level of attention from a cognitive point of view (McPherson, 2006). Empirical data on the effect of piano methods on learning are scarce, and very difficult to obtain in a controlled setting. As a first step to characterise the effect of pedagogical methods on novice performance and learning, a series of computer simulations were designed. The main objective of the simulations was to compare the resulting states of a common cognitive model after learning to play sequences of short piano pieces from different piano methods. The simulations focused on learning the association between the musical notation and the correct motor movements on the piano keyboard. The task to be performed by the model was a form of sight-reading task (Fourie, 2004). The task was to read a note on a music score, and play it on the piano. The model did not intend to capture looking ahead behaviour (Fourie, 2004), the representation and processing of musical sounds (Chikhaoui, Pigot, Beaudoin, Pratte, Bellefeuille & Laudares, 2009), learning motor skills (Jabusch, Alpers, Kopiez, Vauth & Altenmüller, 2009), movement preparation (Palmer, 2005), and multitasking of music reading and motor movements as threaded cognitive tasks (Salvucci & Taatgen, 2008) were excluded from the models.

The Middle-C and Intervallic approaches

This research's focus is to study the possible impact of the different teaching approaches on the acquisition of initial reading skills. By using cognitive modeling, we are hoping to observe through computer simulation the problem solving and decision-making tasks involved in decoding a simple musical score. We want to examine how the different reading systems impact on the perceptual and motor processes. Since the Middle-C approach and the Intervallic approach have dominated the market for many decades now, we have selected two tutor series that are a good representation of each approach: *The A.B.C. of Piano Playing: An Easy Method for Beginners* (Berlin, Koniček & Precious, rev. ed. 1983; original ed. 1941); *The Music Tree: A Plan for Musical Growth at the Piano* (Clark, Goss & Holland, rev. ed. 2000; original ed. 1973; Clark first introduced the intervallic approach under the title *Time to Begin* in 1955). These authors published their first tutor in the middle of the 20th century, both publications have gone through revision and re-edition and both are still in use by piano teachers. In order to understand the basic characteristics of the reading process involved in each approach, a quick overview of their reading system will be provided.

According to Lomax (1990), the Middle-C reading approach became influential in the early 1900s. Introduced by Mathews in *Standard Graded Course of Studies for the Pianoforte in Ten Grades* (1892), it was then popularised by the very successful tutors written by John Thompson *Teaching Little Fingers to Play* (1936) and the *Modern*

Course for Piano (1936). Berlin's *A.B.C. of Piano Playing* (1941) published a few years later and selected for our analysis was very much in line with the earlier Middle-C tutors. This reading approach requires the student to place the thumbs of each hand on middle C. The entire first piece is often played with that note only, and then on the following pieces, one note above and one note below middle C are introduced. As new notes are introduced, note names and traditional staff notation are learned simultaneously. The hand position with both thumbs sharing middle C and the other fingers resting on the surrounding white keys is maintained generally for quite a long period of time so that the student becomes familiar with these notes. This reading approach was extremely influential throughout the second half of the 20th century, Schaum and Cupp (1985) wrote that "the Middle C approach continues to prevail because of its unparalleled success and thoroughness. It is probably the most widely accepted keyboard teaching system presently in use" (p. 68) and Lomax (1990) was affirming "the Middle C Method is still one of the most widely used approaches today" (p. 101).

In 1955, Frances Clark revolutionised the way that music reading could be thought with the publication of her Intervallic approach tutor *Time to Begin*. Elements of this approach had been introduced earlier: partial-staff notation in *Loomis' Progressive Music Lessons* (Loomis, 1875) and the Landmark approach in *Year by Year Books* (Williams, 1924). However, Clark was able to define the Intervallic approach like no one had done before her and she popularised it among piano teachers. She developed a reading system where piano students are taught to read music by recognizing intervals. As Uszler (1991) explains "the Intervallic approach stressed the development of spatial-directional reading habits connected with the formation of hand-shapes and movements that follow from intervallic recognition" (p. 107). Students are encouraged to read by contour recognition and the musical staff is introduced one line at a time. They are thought to recognize steps (neighbouring keys) and skips (skipping over one key) on a partial staff, then intervals are introduced (seconds, thirds, fourths, etc.) and finally they are given certain landmarks on the keyboard and they are thought to distinguish the direction of the music through intervals that are related to these guide posts. Unlike the Middle C approach, the Intervallic approach reinforces playing all over the keyboard.

Simulation of Early Music Reading Skills Acquisition

This section presents the simulation methodology and simulation results obtained by running an initial cognitive model playing a series of musical staves belonging to either the Middle-C or the Intervallic piano methods. The ACT-R cognitive architecture was used to run the simulation (Anderson, Bothell, Byrne, Douglass, Lebiere & Qin, 2004). The simulation procedures consisted of: a) developing an initial cognitive model, b) running the cognitive model with

the different conditions represented by the different sequence of music staves from the two piano methods, and c) comparing the model states resulting from the separate simulations.

Initial cognitive model

The initial model contained only the minimal declarative and procedural knowledge to be able to visually scan a music staff for notes, the piano keyboard for keys, move the hands and fingers over piano keys, press, hold and release them, and the capabilities to process instructions from a tutor. In addition to the content of the declarative and procedural memories described in the following sections, the cognitive model also used base level activation of declarative chunks, production rules compilation, and reinforcement learning.

Declarative knowledge. The initial model assumed no prior knowledge of musical notation, and of its association to specific key locations on the piano keyboard. The only declarative knowledge the initial model held were chunks about the association between the number of beats (1 to 4), and the subjective perception of time encoded as ticks. The model however had chunks encoding the approximate duration of 1, 2, 3, and 4 beats (60 beats per minute) using the ACT-R temporal module (Taatgen, van Rijn & Anderson, 2004).

The Figure 1a and 1b presents the visual encoding of the music scores. As figure shows, both the Middle-C and the Intervallic methods share the same encoding, in spite of the differences in the layouts. The visual encoding of a note visual location includes its X and Y absolute visual locations, its relative horizontal and vertical visual locations, as well as four duration encodings using a combination of full or empty circles, with or without stems, and with or without a dot.

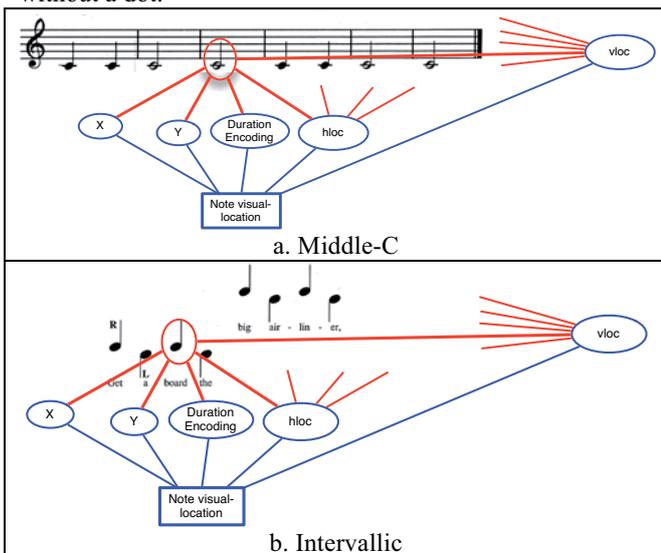


Figure 1: ACT-R visual encoding of music staves.

The Figure 2 presents the visual encoding of the piano keyboard. This encoding is used to direct the hands towards

the proper key to associate with the encoding of the note information on the music staff. The visual encoding of a key location includes the absolute X and Y visual locations, the key color (black or white), the group type (around 2 blacks or 3 blacks), the relative position of a key in the group, as well as the relative position of the group on the keyboard.

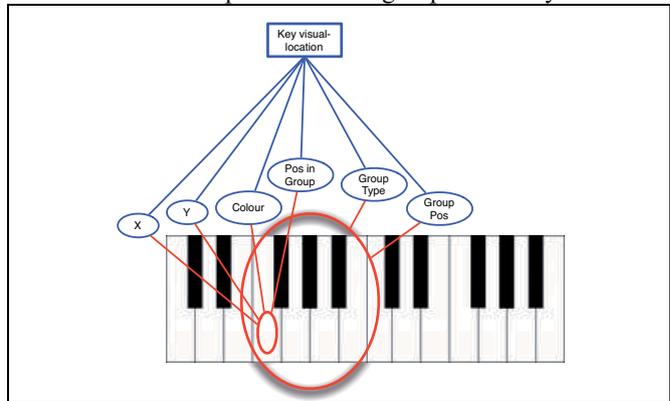


Figure 2: Visual encoding of the piano keyboard using ACT-R chunks.

In addition to the visual encoding of the staves and the keyboard, the model includes a chunk type representing the knowledge about a note, which binds together the musical notation information (staff, vertical location on the staff, duration encoding), motor directives (number of beats, hand, and finger to use), and associated key on the keyboard (group type, group position, key position in group, and key colour). This representation aims at capturing the visual characteristics of notes for musical notations, and in this respect, it differs from a representation of its sound properties (Chikhaoui et al., 2009).

Closely related to the note chunk, the model includes an execution plan. An execution plan is basically a note chunk augmented with the information about the horizontal position of a note on the staff to encode the sequence of notes to play, and the number of ticks (Taatgen et al., 2004) that the note should be pressed. The execution plan acts as the control structure for the model's behaviour. Chunk slots are filled up based on visual encoding and memory retrievals until the plan can be executed. Plan execution chunks are held in the goal buffer of the ACT-R cognitive architecture. The encoding for the note is similar to the theory of event coding where perception and action share a common representation (Hommel, Müsseler, Aschersleben & Prinz, 2001).

Procedural knowledge A total of 19 productions are part of the model's initial procedural knowledge. These productions can be classified in productions for processing the tutor's instructions (2), processing the visual information on the staff (2), determining the note duration (5), its key location on the keyboard (4), the finger and hand to use (4), and finally executing the motor action on the keyboard (2). The Figure 5 characterizes the overall flow of control in the model. The first task of the model is to attend the staff and encode the next note visual features. Then the model

attempts to retrieve from declarative memory a note chunk using the visual features as cues. The retrieved note chunk slots are used (or guessed if no note is retrieved) to complete the missing information in the execution plan. The note duration, fingering and key location need to be determined in no particular order. Once the execution plan is completed, the model locates the key on the keyboard, move the hand and finger to the location, and press and hold the key for the given duration.

Figure 3 also includes a description of the flow of control between the student model and an automated tutor. The tutor compares the note to be played by the student model to its performance and provide either a positive reward, or a negative reward with instructions. An instruction consists of a note chunk, correcting the note played. After the reception of an instruction, the model harvests its content to declarative memory, and proceeds to re-attend to same note on the staff. If the note played was correct, the model just proceeds to the next note on the staff.

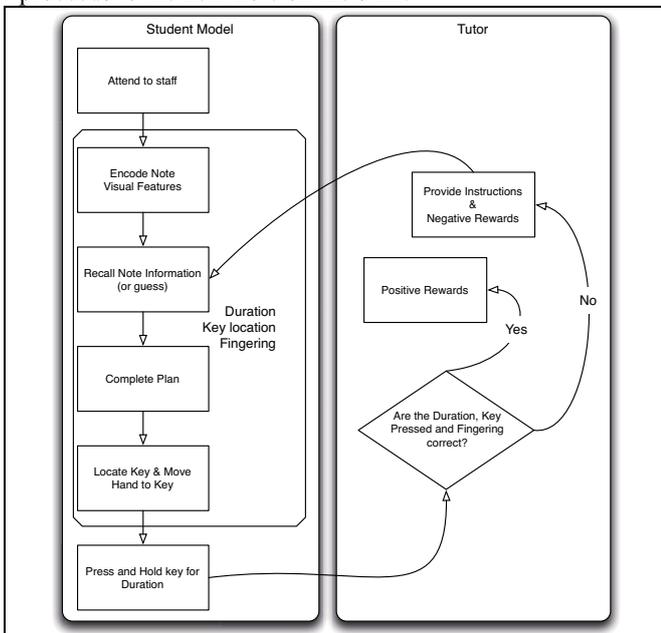


Figure 3: Flow of control and interaction with tutor.

Running the simulation

The simulation consisted of running a sequence of introductory piano pieces from the Middle-C method, and another one from the Intervallic method. For both sequences, the model started in an identical initial state (described in the previous section). Each sequence had 8 pieces and the model had to play every piece 5 times before moving to the next piece. The following pieces were used in the Middle-C and Intervallic conditions.

Middle-C (Berlin et al., 1983): Second lesson right, Second lesson left, third lesson right, third lesson left, fourth lesson right, fourth lesson left, sixth lesson right, sixth lesson left.

Intervallic (Clark et al., 2000): Take Off, Landing, In a Canoe, Space Ship Inchworm, Rock Band, On the Bleachers, Halloween.

After each executed pieces, model states data were collected, in particular the number of declarative chunks in memory, as well as the trace of production rules execution, and their relative utility.

Results and discussion

There types of data were collected during the simulation execution: the number of declarative chunks in memory, the trace of production rules execution, and their relative utility. The aggregated results are presented in the Figures 4 and 5.

Figure 4 shows the number of declarative chunks in memory as the model progress through the execution of the 40 pieces of music (8 different pieces played 5 times). As the graphic shows, the Middle-C method (lower line) has a very gradual introduction of musical note information when compared to the Intervallic method. The main reason for this difference is somewhat obvious. Because the intervallic method forces the learning musician to play over multiple octaves, the number of note chunks is therefore larger, reflecting the demands of the music scores.

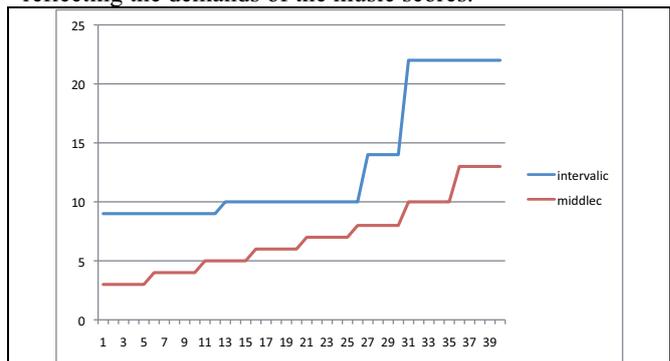


Figure 4: Number of declarative chunks as a function of pieces played.

Figure 5 shows the percent of time spent by the model on building an execution plan, which means the exclusion of the time devoted to visual encoding and motor execution, and the inclusion of processes related to instruction encoding, retrieval, and filling up the execution planning chunk slots. A visual inspection of the graph seems to indicate that the Middle-C method (lower line) requires less retrieval and execution planning time than the Intervallic method. Similar to the previous result on the number of declarative chunks, the larger number of notes to be played with the Intervallic method demands more motor planning. However, the line threads seem to also have different patterns. The Intervallic method has more or less a constant planning time over the course of the simulation. On the other hand, the Middle-C method seems to require an increase of planning time. This increase could be correlated with the increase of notes in the method. The apparent consistency of planning time for the Intervallic method might reflect a ceiling effect cause by the constant number of features per note (location, duration, fingering).

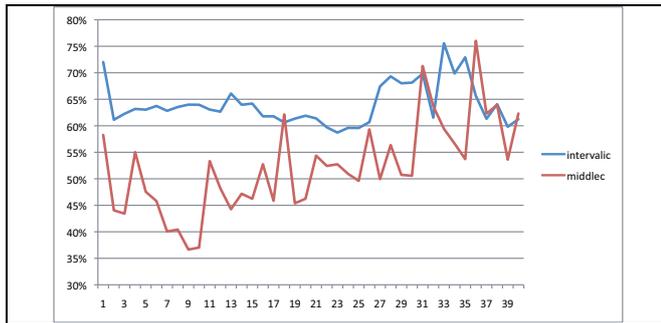


Figure 5: Percent of time spent on building an execution plan.

Results from the production compilation indicated that the model learnt to skip productions, reflecting knowledge acquired about the meaning of the notes. Both methods generated similar productions and their utility values were comparable. For both piano methods, the utility values of new productions were larger than the initial production utilities, in particular for the productions related to the note information associated to the plan duration of a pressed keyboard note.

Conclusion

Advanced music reading skills (sight-reading) exhibits a smooth coordination of visual encoding and motor skills (Fourie, 2004; Kopiez & Lee, 2008). With skill development, this combination requires a transition from multitasking to cognitive processes concurrency. As notes are being read on the staff, motor movements are planned and executed, while the reading process is progressing beyond what is currently played. Sight-reading efficiency demands the coordination of psycho-motor speed, early acquired expertise, mental speed, and the ability for auditory imagery (Kopiez & Lee, 2008).

As an initial step towards characterizing the effect of different piano methods on the acquisition of piano playing skills, we constructed a minimal cognitive model which acquired declarative and procedural knowledge through the execution of novice piano pieces from the Middle-C and Intervallic methods. Inspection of the resulting models revealed differences in terms of declarative memory and cognitive processing demands. In particular, the intervallic method requires a larger number of declarative knowledge related to notes, and more gesture planning than the Middle-C method.

There are some limitations to the current state of the research that need to be mentioned. In particular the model would need to integrate a representation of sound to a note (Chikhaoui et al., 2009). This is important because the inner playing of a piece of music is a good determinant of music reading performance (Fourie, 2004). Also the model only focuses on individual note and has no notion of musical phrase. A more realistic model of motor movement could also be added, but mostly the model should be able to address the visual and motor concurrency and the development of reading ahead strategies. The model does not aim at

modelling errors. For example Fourie (2004) reports that 80% of error in sight-reading are rhythmic in nature, probably caused by the difficulty related to locating the correct key on the keyboard. This measure could be an interesting one in comparing the Middle-C and Intervallic methods, given the larger number of keyboard keys in the latter method. In this respect, the model should also have a representation of intervals, which as the moment is not present. Note accents were left out of the simulation, even though it is present in the introductory pieces of both the Middle-C and Intervallic methods.

Acknowledgments

We would like to thank the Social Sciences and Humanities Research Council for their financial support.

References

- Altenmüller, E., Wiesendanger, M., & Kesselring, J. (2006). *Music, motor control and the brain*. New York, NY: Oxford University Press.
- Anderson, J. N. (1981). Effects of tape-recorded aural models on sight-reading and performance skills. *Journal of Research in Music Education*, 20(1), 23-30.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, 11(4), 1036-1060.
- Berlin, B. (1941). *The A.B.C. of piano playing: An easy method for beginners*. Mississauga, ON: The Frederick Harris Music Company.
- Berlin, B., Koniček, L., & Precious, C. (1983). *The A.B.C. of piano playing: An easy method for beginners*. Oakville, ON: The Frederick Harris Music Company.
- Chikhaoui, B., Pigot, H., Beaudoin, M., Pratte, G., Bellefeuille, P., & Laudaes, F. (2009). *Learning a song: An ACT-R Model*. Proceedings of the International Conference on Computational Intelligence (pp. 405-410). Oslo, Norway.
- Clark, F., & Goss, L. (1973). *The music tree: A plan for musical growth at the piano*. Princeton, NY: Summy-Birchard Inc.
- Clark, F., Goss, L., & Holland, S. (2000). *The music tree: A plan for musical growth at the piano*. Miami, FL: Summy-Birchard Inc.
- Clark, F., & Goss, L. (1955). *Time to begin*. Evanston, IL: Summy-Birchard Inc.
- Fourie, E. (2004). The processing of music notation: Some implications for piano sight-reading. *Journal of the Musical Arts in Africa*, 1, 1-23.
- Galyen, S. D. (2005). Sight-reading ability in wind and percussion students: A review of recent literature. *UPDATE Application of Research in Music Education*, 24(1), 57-70.
- Hahn, L. B. (1985). Correlations between reading music and reading language, with implications for music

- instruction (notation). (University of Arizona, 1985). *Dissertation Abstract International*, 46(09), AAT 8525597.
- Hallam, S., Cross, I., & Thaut, M. (2009). *The Oxford handbook of music psychology*. New York, NY: Oxford University Press.
- Hodges, D. A. (1992). The acquisition of music reading skills. In R. Colwell (Ed.), *Handbook of research in music teaching and learning* (pp. 466-471). New York: Schirmer Books.
- Hodges, D. A. (Ed.) (1996). *The Handbook of music psychology* (2nd ed.). St. Louis, MO: MMB Music.
- Hommel, B., Müsseler, J., Aschersleben, G., & Prinz, W. (2001). The Theory of Event Coding (TEC): A framework for perception and action planning. *Behavioral and Brain Sciences*, 24, 849-937.
- Jabusch, H. C., Alpers, H., Kopiez, R., Vauth, H., & Altenmüller, E. (2009). The influence of practice on the development of motor skills in pianists: A longitudinal study in a selected motor task. *Human Movement Science*, 28, 74-84.
- Kinsler, V. & Carpenter, R.H.S. (1995). Saccadic eye movements while reading music. *Vision Research*, 35(1), 51-65.
- Kopiez, R., & Lee, J. I. (2008). Towards a general model of skills involved in sight reading music. *Music Education Research*, 10(1), 41-62.
- Lemay, C. (2008). The measurement, quantification and evaluation of piano students' music reading performance: Comparing methods of assessment (Master theses). Retrieved from ProQuest Digital Dissertations & Theses database.
- Lomax, E. (1990). *A comparison of three approaches to teach note-reading and note location on the piano keyboard to children, ages four to six*. Retrieved from Proquest Digital Dissertations. (AAT 9035066)
- Loomis, G. (1875). *Progressive music lessons: A course of instruction prepared for the use of public schools*. New York, NY: Ivison, Blakeman, Taylor.
- Madell, J., & Hebert, S. 2008. Eye movements and music reading: Where do we look next? *Music Perception*, 26(2), 157-170.
- Mathews, W. S. B. (1892). *Standard graded course of studies for the pianoforte in ten grades: Consisting of standard etudes and studies arranged in progressive order; selected from the best composers for the cultivation of technic, taste and sight reading; carefully edited and annotated and supplemented with complete directions for the application of mason's system of technics in each grade for the production of a modern style of playing*. Philadelphia, PA: Theodore Presser.
- McPherson, G. E. (2006). *The child as musician. A handbook of musical development*. New York, NY: Oxford University Press.
- McPherson, G. E., & Gabrielsson, A. (2002). From sound to sign. In R. Parncutt and G. E. McPherson (Eds.), *The science & psychology of music performance: Creative strategies for teaching and learning* (pp. 99-115). New York, NY: Oxford University Press.
- Palmer, C. (2005). Time course of retrieval and movement preparation in music performance. *Annals of the New York Academy of Sciences*, 1060, 360-367.
- Parncutt, R., & McPherson, G. E. (2002). *The science & psychology of music performance*. New York, NY: Oxford University Press.
- Salvucci, D. D., & Taatgen, N. A. (2008). Threaded cognition: An integrated theory of concurrent multitasking. *Psychological Review*, 115, 101-130.
- Schaum, W., & Cupp, J. (1985). *Keyboard teaching with greater success and satisfaction*. New York, NY: Schuam Publications.
- Sloboda, J. (1978). The psychology of music reading. *Psychology of Music*, 6, 3-20.
- Sloboda, J. (1985). *The Musical Mind. The cognitive psychology of music*. NY: Oxford University Press.
- Sloboda, J. (2005). *Exploring the musical mind: Cognition, emotion, ability, function*. New York, NY: Oxford University Press.
- Stewart, L., Henson, R., Kampe, K., Walsh, V., Turner, R., & Frith, U. (2003). Brain changes after learning to read and play music. *Neuroimage*, 20(1), 71-83.
- Taatgen, N., van Rijn, H., & Anderson, J. R. (2004). *Time perception: Beyond simple interval estimation*. Presented at the Sixth International Conference on Cognitive Modeling (pp. 296-301). Pittsburgh, PA.
- Thompson, J. (1936). *John Thompson's modern course for the piano*. Cincinnati, OH: Willis Music.
- Thompson, J. (1936). *John Thompson's Teaching little fingers to play*. Cincinnati, OH: Willis Music.
- Udtaisuk, D., B. (2005). A theoretical model of piano sightplaying components (Doctoral dissertation). Retrieved from ProQuest Dissertations & Theses database. (UMI No. 3204275).
- Uszler, M., Gordon, S., & Mach, E. (1991). *The well-tempered keyboard teacher*. New York, NY: Schirmer Books.
- Uszler, M., Gordon, S., & Smith, S.M. (2000). *The well-tempered keyboard teacher*. Belmont, CA: Schirmer Books.
- Waters, A.J., Townsend, E. & Underwood G. (1998). Expertise in musical sight reading: A study of pianists. *British Journal of Psychology*, 89, 123-149.
- Williams, J. (1924). *Year by year at the piano: A progressive and modern beginner's book*. Philadelphia, PA: Theodore Presser.
- Wolf, T. (1976). A cognitive model of musical sight-reading. *Journal of Psycholinguistic Research*, 5(2), 143-171.

An account of cognitive flexibility and inflexibility for a complex dynamic task

Alberto De Obeso Orendain (a.de-obeso-orendain@sussex.ac.uk)
Cognitive & Languages Processing Systems Group, School of Informatics
University of Sussex, Falmer, Brighton, BN1 9QJ, UK

Sharon Wood (s.wood@sussex.ac.uk)
Cognitive & Languages Processing Systems Group, School of Informatics
University of Sussex, Falmer, Brighton, BN1 9QJ, UK

Abstract

Problem solving involves adapting known problem solving methods and strategies to the task at hand (Schunn & Reder, 2001) and cognitive flexibility is considered to be “the human ability to adapt the cognitive processing strategies to face new and unexpected conditions of the environment” (Cañas et al., 2005, p. 95). This work presents an ACT-R 6.0 model of complex problem solving behavior for the dynamic microworld game FireChief (Omodei & Wearing, 1995) that models the performance of participants predisposed to behave either more or less flexibly based on the nature of previous training on the task (Cañas et al., 2005). The model exhibits a greater or lesser degree of cognitive inflexibility in problem solving strategy choice reflecting variations in task training. The model provides an explanation of dynamic task performance compatible with the Competing Strategies paradigm (Taatgen et al., 2006) by creating a second layer of strategy competition that renders it more flexible with respect to strategy learning, and provides an explanation of cognitive inflexibility based on reward mechanisms.

Keywords: complex problem solving; cognitive inflexibility; dynamic tasks; strategy use; adaptation.

Introduction

Problem solving involves adapting known problem solving methods and strategies to the task at hand (Schunn & Reder, 2001) and cognitive flexibility is considered to be “the human ability to adapt the cognitive processing strategies to face new and unexpected conditions of the environment” (Cañas et al., 2005, p. 95). When approaching a new problem, it is thought that problem solvers with higher levels of cognitive flexibility will outperform those who are less flexible because the former tend to consider alternative ways to solve the problem (Stewin & Anderson, 1974) rather than rigidly adhering to well-used methods. In their study of cognitive flexibility, Cañas et al. (2005) found that participants became predisposed to behave either more or less flexibly based on the nature of previous training on the task. Those trained repeatedly on the same problem scenario developed a preference for how they solved the task, becoming faster and more fluid in their actions over time. When subsequently tested on a different scenario their behavior was inflexible in adapting to the new test conditions and performance suffered. In contrast, those trained on a series of varying problem solving scenarios demonstrated an ability to adapt their problem solving behavior flexibly to the challenges presented by the new test scenario. The work presented here describes an ACT-R

model for the Cañas et al. (2005) problem solving task that demonstrates varying degrees of cognitive flexibility depending on the training regime it undergoes. Analysis of the model provides an explanation of cognitive inflexibility based on reward mechanisms.

Background

There are several cognitive modeling paradigms (Taatgen et al., 2006) for problem solving involving strategy selection. In the Competing Strategies paradigm (ibid.), several strategies are implemented in a cognitive architecture and then compete for use in solving a problem. According to Taatgen et al. (2006) utility learning can be used to determine the best strategy. This paradigm has been successfully applied in modeling problem solving behavior for static tasks (Lovett & Anderson, 1996; Peebles & Bothell, 2004) and tasks in dynamically changing situations such as Air Traffic Control (Schunn & Reder, 2001; Schoelles & Gray, 2000).

Dynamic problem solving tasks pose an added layer of complexity. In dynamic situations the problem solver needs to execute not only the appropriate action but also to implement it at the right time: a good decision at one moment could be ineffective the next. In order to obtain good performance both selection and execution of the chosen strategy must be effective.

Problem solvers must also be ready to change strategy as and when the situation demands (Gonzalez et al., 2004); they must continuously process feedback in order to select appropriate actions within an ever-changing situation (Brehmer & Dörner, 1993). Underlying this ability, according to Schunn & Reder (2001), strategy choice is influenced by overall success and “Dynamic tasks bring to the forefront the importance of the ability to adapt to changing success rates” (p. 61). They argue that although participants may use a similar set of strategies they can differ in their ability to opportunistically apply those strategies in response to the situation.

This ability to adapt behavior may be affected by factors such as cognitive inflexibility, which can be produced as a consequence of the way problem solvers interact with the task at hand. As skill in a task improves and becomes more automatic so cognitive inflexibility may increase, particularly in tasks with a high level of consistency (Ackerman, 1988). For example, in a fire-fighting task, Cañas et al. (2005) found evidence of cognitive inflexibility in participants trained repeatedly on the same problem scenario who, having found an effective strategy, failed to

relinquish it despite situational changes that reduced its effectiveness. This contrasted with participants trained on a variety of different problem scenarios.

However, studies investigating cognitive inflexibility have not always drawn consistent results. For example, Schunn & Reder (2001) found no evidence for cognitive inflexibility in their study involving training on an Air Traffic Control task when situational changes affecting success on the task were introduced.

The work presented here implements an ACT-R model of the Cañas et al. (2005) study to elucidate the mechanisms of cognitive inflexibility further in an attempt to reconcile these disparate findings.

The FireChief Microworld

The Cañas et al. (2005) study used a dynamic microworld game called FireChief (Omodei & Wearing, 1995) for the problem solving task. Figure 1 shows the FireChief display.

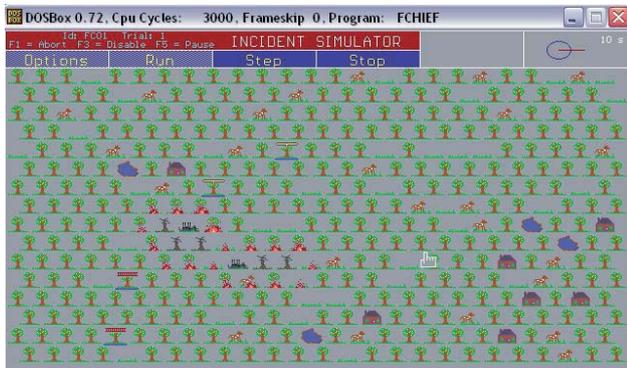


Figure 1: The FireChief microworld display

Players combat fires spreading in a landscape using truck and copter fire-fighting units. A FireChief problem scenario depicts a landscape comprising forest, clearings and property, the position of initial fires, fire-fighting units, and the direction and strength of the wind. Copter and trucks can be moved between landscape grid cells and Drop Water (DW) over cells to extinguish fires. Copters move three times faster than trucks and cannot be destroyed by fire, but a truck's water tanks have double capacity and are able to Control Fire (CF) by creating a fire-break. Commands are issued through a combination of mouse and keyboard operations and their execution takes a fixed amount of time (4 seconds to DW; 2 seconds to CF) and a variable amount of time to Move a unit depending on distance and type of unit. Wind strength and direction are in the upper right-hand corner of the display. Task performance is inversely proportional to the number of cells destroyed by fire at the end of the trial.

The FireChief problem state changes both independently and as a consequence of the participant's actions and time pressure is directly related to fire development, which depends heavily on wind strength.

The Cañas et al. (2005) study

Each trial for the FireChief task lasts 260 seconds. The experimental data comprises a list of commands executed

during each trial that is indexed to a detailed description of the changing scenario. The first 16 trials comprise the training phase and the last 8 trials the testing phase. There were two training conditions: constant and variable.

In the constant training (CT) condition the problem scenario is exactly the same for each trial and wind strength and direction remains fixed. In the case of variable training (VT) a different scenario is presented in each of the sixteen trials. Trials vary in landscape composition, initial position of fire-fighting units and fires and, importantly, wind direction and strength varies throughout the trial.

There are also two test conditions. In the Wind Direction Change (WDC) condition the wind changes direction every 60 seconds. These shifts in wind direction have a dramatic impact on fire development. In the second Efficiency Reduction (ER) test condition, appliances deliver less water and are therefore less effective in extinguishing fires.

As previously hypothesized, Cañas et al. (2005) found participants in the CT condition improved performance as the number of trials increased; however, during the test phase this same group demonstrated a distinct lack of flexibility in adapting their problem solving strategy to the new task demands. In contrast, participants in the VT condition demonstrated a greater facility for changing strategies under test conditions. The findings were consistent across both WDC and ER test conditions

The Model

The ACT-R 6.0 (Anderson et al., 2004) model interfaces to a LISP version of the FireChief microworld (De Obeso Orendain & Wood, 2010). Task knowledge comprises both procedural (condition-action) rules that produce behavior according to four high level strategies: *Barrier*, *Non-Barrier*, *Stop*, and *Follow* (ibid.) and three declarative knowledge components that impact this behavior: (1) the *goal chunk*, the main task objective is to extinguish the fire; (2) the *strategy specification chunk*, which defines whether the model will use a mixture of DW and CF commands, whether or not a barrier will be created, and which method of attacking the fire is preferred (attack weak fires, attack strong fires or attack the strongest fire); and (3) the *intention chunk*, used to track the current intention (stored in the ACT-R imaginal buffer, Anderson et al., 2004). Intentions emanate from steps in pursuit of the main goal, according to the chosen strategy.

The model identifies its preferred strategy by comparing the utility of its four strategy rules, combined with a situation assessment, and retrieves the corresponding *strategy specification chunk*. This chunk remains unaltered throughout the entire trial, unless there is a strategy change.

Overall the model behavior reflects the use of procedural knowledge over declarative knowledge: it is constructed in such a way that it is mainly controlled by the utility learning mechanism. The content of the three declarative chunks determine which rules are applicable in different situations, but there is always more than one eligible rule, so the decision about what to do next is taken in terms of utility.

ACT-R's utility learning mechanism

Utility designates the perceived value of implementing a procedural rule, and thereby its associated behavior, and is updated via a reward mechanism reflecting task success. Throughout runtime, Rule utilities are compared during the process of conflict resolution where only the rule with the highest utility is selected and thereby acted upon. In ACT-R when a reward is triggered the utility values of all rules that have fired since the last reward are updated. The actual reward allocated depends on the absolute value of the reward and the length, in time, between the giving of the reward and the execution of that rule. The consolidation of strategies and the existence of cognitive inflexibility discussed here are explained in terms of utility variations in the set of rules identified as key in implementing a strategy. A key rule is one that enters the conflict set during ACT-R conflict resolution and hence competes in determining the next intention or action of the model.

Achieving adaptivity

The considerable variability observed in participants' protocols suggests that for the FireChief task there is variation not only in strategy choice, but also in the chosen method of execution. The dynamic nature of FireChief introduces a dynamic component into the execution of strategies that forces a second layer of competition between alternative courses of action within the same strategy. For this reason a paramount feature of the model is to enable this kind of competition. In the FireChief task there are four fire-fighting units (Copter, Truck), three commands (DW, CF, Move) and four hundred locations. From a very broad perspective the model's operations are devoted to determining the agent, type and spatial location of the next command and a strategy functions as a mechanism for helping the model to constrain this decision. Two types of control coexist within the model. The current representation of the task (the *strategy specification chunk*) guides actions through top-down control. Nevertheless bottom-up control is particularly relevant when considering dynamic tasks therefore feedback from the environment is used to guide the further selection of actions by triggering a wider variety of rules than those specified in the strategy chunk.

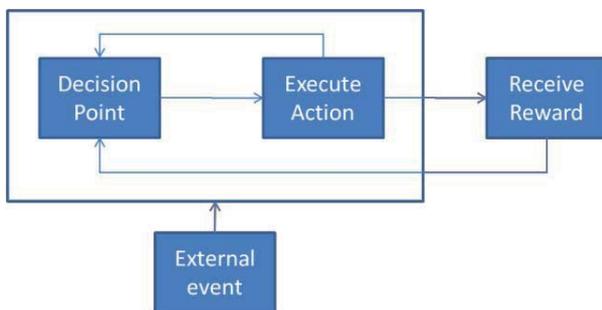


Figure 2: The basic cycle of the model comprising a second layer of within-strategy competition

The model uses an adaptation of the Competing Strategies paradigm (Taatgen et al., 2006): the core of the model is the *Decision Point/Action/Reward* cycle shown in figure 2.

The basic cycle starts with a *Decision Point* (identifying eligible rules) continues with the *Execution* of an action (rule-firing), and finishes with the awarding of a *Reward*. The branching factor at every *Decision Point* is variable and there are *External Events* that can interrupt the flow of actions in the cycle such as alarms and visible changes in the environment that prevent the effects of an action taking place, for example, a cell catching fire before a CF command is completed. The model is designed in such a way that *Decision Points* occur frequently. In this way the model is mainly governed by the utility values of its rules. This bottom-up control feature results in the emergence of interesting behaviors (observed in participants) such as “waiting behavior”: when a truck is Moved to a cell with the intention of issuing a CF command, if the movement's length is shorter than 2 cells, the model tends to wait for the unit to arrive (incurring in a waste of time but increasing the probability of issuing the CF command as soon as the unit arrives, rendering its success more likely). The description and analysis of emergent behaviors is outside the scope of this paper.

The same set of rules is used for modeling performance of the task under both training conditions from the Cañas et al. (2005) study. However, rewards for task performance and thus specific rule utility values will vary according to the unique experience of the model on any given trial (model run). Furthermore, these utility values will accumulate over both training and testing phase.

Rewarding the execution of commands

Within the model positive rewards are received for successfully completing commands and negative rewards for failing to execute commands successfully or for wasting time (this means that the utility of a rule can be negative). In this way, any action that contributes to the successful completion of a command is rewarded predisposing the model to continually issue commands. External reward: final performance

In addition to built-in ACT-R utility learning mechanisms a further external reward mechanism affects the utility of the four strategy rules. The strategy rule invoked for a given trial is modified at the end of each trial based on final performance (the amount of non-destroyed terrain remaining at the end of the trial). For instance, if the rule that selects the *Stop* strategy is fired and the final performance achieved during the trial is high, the rule's utility is increased. Manipulating rule utilities outside the standard ACT-R mechanism, has also been used elsewhere (e.g., Schoelles & Gray, 2000).

Results

Data fitting: The model was fitted to the Cañas et al. (2005) study participant data as described in De Obeso Orendain & Wood (2010).

Performance: During the *training* phase the average performance of participants in the CT group is 78.7 while

the average performance of the model for CT is 77.1. In the VT group, the average performance of participants is 78.45 versus 81.2 for the model. The fit of the model is better for the *Barrier* and *Stop* strategies ($r^2=.987$) which are the most structured strategies (De Obeso Orendain & Wood, 2010).

Strategy use: For the CT *training* scenario the *Barrier* strategy using CF commands to construct a fire-break (ibid.) is a good option because the fire develops quickly and soon reaches an intensity that surpasses the capability of the fire-fighting units. In the CT condition both participants and the model use the *Barrier* strategy increasingly more frequently, by trial 16 participants use the *Barrier* strategy 71% of the time while the model is using it 79% of the time.

Strategy change: During the *training* phase participants in the VT group change strategy with more frequency than participants in the CT group, the model captures this tendency ($r^2=.93$ RMSD=1.43). The fact that both participants and model use the *Barrier* strategy more frequently, and there is less strategy change, during CT facilitates the consolidation of this strategy in the CT group.

Learning in CT: A significant performance increment was obtained by comparing the first (1-4) and last four *training* trials (12-16) for both participants and the model. ($F(1,33)=4.417$, $p<.05$ and $F(1,33)=5.17$ $p<.05$ respectively). This means that consolidating the use of the *Barrier* strategy is beneficial by objective criteria.

Cognitive inflexibility: After the training period both participants and the model undergoing the CT condition exhibit inflexibility on two levels: strategy choice and strategy implementation. Both kinds of inflexibility can be traced to variations in key rule utility values induced by the two training conditions.

The set of rules available for use are exactly the same for both training conditions (a single model undergoes either of the training conditions). However, the pattern of change in utility values varies as a consequence of the training received. As shown in figure 3 for the *Barrier* strategy: over the sixteen training trials average utility values of *Barrier* strategy rules for the CT group (TOP-DOWN CT) far exceed those for the VT group (TOP-DOWN VT).

This contributes towards an explanation of cognitive inflexibility in strategy choice. As a consequence of the CT condition, the reward function shapes the utility values of the model's rules in such a way that it becomes relatively insensitive to changes in reward. The high utility values of rules for the preferred *Barrier* strategy in the CT group shield the model from relatively small variations in success. When creating a barrier is no longer the best approach, such as occurs during the test phase, the model will eventually change its behavior through repeated negative reward after the utility values of the rules for the preferred strategy have reduced sufficiently in comparison to the rules for alternative strategies. But this takes time, giving rise to the observable phenomenon of cognitive inflexibility.

In contrast, the model subjected to the VT condition is more sensitive to changes in reward during the test phase because its rules for implementing alternative strategies are more evenly weighted; because the differences between their utility values is smaller, a small amount of negative reward is able to trigger a switch to an alternative strategy.

Differences in utility also contribute towards an explanation of cognitive inflexibility in strategy implementation, again discussed here in relation to the *Barrier* strategy.

There are a range of actions that might be involved in constructing a barrier by a variety of methods represented as the set of rules available whenever the *Barrier* strategy is selected. One subset of rules comprises methods that implement the *Barrier* strategy in a structured top-down manner. For example, top-down *Barrier* strategy rules systematically identify the next section of the barrier to be constructed by locating CF commands in grid cells adjacent to that section of the barrier just formed.

In comparison, other rules involve a greater degree of bottom-up control in implementing actions. For example, a bottom-up strategy rule might locate the next section of the barrier to be constructed by looking to see where the fire is before making a decision about where to put the next section of the barrier. These top-down and bottom-up rules compete throughout the creation of a barrier (while the *Barrier* strategy is selected) and those selected by ACT-R give rise to the final form of the barrier.

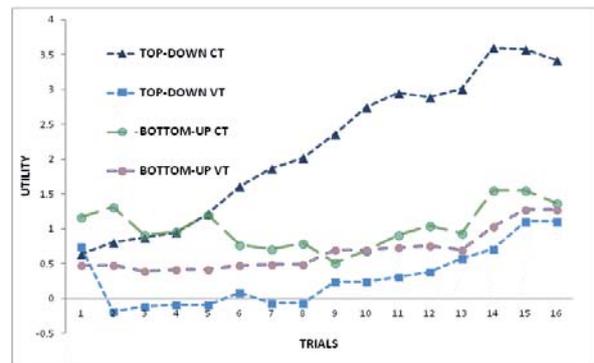


Figure 3: Changes in top-down strategy and bottom-up responsive *Barrier* rule utilities during training

Figure 3 shows the average utility values for these two notional subsets of rules over the sixteen *training* trials: the utility of the rules implementing the strategy top-down increases as more trials are completed during CT (TOP-DOWN CT) as their repeated use is continuously rewarded. This phenomenon occurs only when the problem scenario does not vary dramatically between trials so that there is no significant variation in the effectiveness (and thus reward value) of the actions being executed on repeated trials. In comparison the bottom-up responsive rules involve many more perceptual actions to locate the spread of fire, taking longer to construct the barrier, consequently receiving a relatively lower reward (BOTTOM-UP CT). Over time, this serves to increase the probability of using the top-down subset of rules in the CT group producing the divergence shown in Figure 3. The utility values for the same notional subsets of rules for the VT group, again, remain more evenly balanced owing to the variability in training rewarding the top down implementation of the strategy less consistently.

As in the case of strategy choice, CT leads to cognitive inflexibility in strategy implementation, with potentially insufficient regard given to sensing the environment over top-down construction of the barrier, when conditions change, as witnessed for the CT group under test conditions.

Testing phase: Comparisons were made to determine the impact of cognitive inflexibility on performance in the first *testing* trial. The average performance in the 17th trial in the ER condition is better for participants/model after CT (86.09/78.14) than after VT (72.19/69.83). Both participants and the model in the CT group use the *Barrier* strategy more effectively than the VT group in the ER condition, an indication that these participants have consolidated the Barrier strategy following a top-down approach. The CT group does not need to change strategy because using CF commands is the only sure way to stop the fire in the ER condition (and constructing a barrier using CF is the best approach and therefore has an advantage). The average performance in the 17th trial in the WDC condition is better for participants/model after VT (78.51/78.14) than after CT (71.38/74.87). This is because shifts in wind direction make fire behavior unpredictable so flexible behavior is required. This flexibility is best achieved using more situation-sensitive responsive rules such as those contributing more bottom-up control in the creation of the barrier.

Control of behavior: Figure 3 shows that the model trained in the CT condition has a clear preference for the use of top-down control while the model trained in the VT condition has no such preference. This difference has an impact in the WDC test phase when the wind changes direction in trial 17 at second 60. In a model trained in the VT condition the bottom-up rules are more easily able to win the competition through small variations in utility values following negative reward. Therefore, when the change in the wind occurs, the model will probably select the next target cell based on the location of the fire. On the other hand, the behavior of a model trained in the CT condition will reflect its high utility rules implementing the top-down approach to the creation of the barrier so it will continue to place the next section of barrier without recourse to observing the fire. The risk is that when the form of the barrier is constructed without considering the actual shape of the fire it may not be effective. In this sense the *automation* of the strategy (cf. Ackerman, 1988) runs the risk of deterring the problem solver from extracting relevant information about the problem state to guide behavior.

To validate the results obtained with the cognitive model, further evidence to support this interpretation was sought from the spatial distribution of CF commands in the Cañas et al. (2005) study data for participants during the WDC testing phase: groups CT-WDC and VT-WDC to determine whether the semicircle pattern, a top-down control outcome, was present. These test groups were chosen because the wind direction change test condition alters the path of the fire in such a way as to make the top-down control implementation of the Barrier strategy less effective than a bottom-up more responsive mode of barrier construction. It was found that the CT-WDC group data presents a semicircle pattern of barrier, evidence of top-down

application of the *Barrier* strategy, whilst the VT-WDC group does not. This indicates that the semicircular pattern does not emerge in the VT group behavior because the variability of both the VT condition and the WDC testing phase does not reward the rules implementing it.

Discussion

The model captures the behavior of both training groups with a single set of rules for implementing all four strategies either or both bottom-up and top-down control. Participants in the CT group have the opportunity to consolidate their strategies and hence generate quick, fluid actions; while those in the VT group execute more controlled, albeit flexible, actions. When the testing phase begins people in the CT group are less (cognitively) flexible in adapting to the new demands of the task. In general terms, participants in the VT condition changed strategy more often and showed more cognitive flexibility during the testing phase. The model demonstrates how cognitive inflexibility can be traced to the utility values of rules governing behavior indicating the potential role of reward feedback learning mechanisms in complex problem solving in dynamic domains.

The CT condition presents to the model more stable feedback from the environment (in the form of rewards) to its actions in comparison with the VT condition. In the CT condition the model tends to respond by executing CF commands in a fashion that resembles a barrier. As experience in the task is gained, the model learns how to deploy this strategy with more efficiency.

The ACT-R reinforcement learning mechanism is able to capture the phenomenon of cognitive inflexibility but in order to achieve this it was necessary to provide the model with adequate responsiveness. Rather than following a recipe to implement a strategy, the approach used in this research was the *Decision Point/Action/Reward* cycle which (using standard ACT-R mechanisms) maximizes the number of decision points during strategy execution and thereby enforces competition between rules in selecting the next action at almost every time step so that the model can find the best way of implementing a strategy. This reflects the model's dependence on ACT-R's sub-symbolic processes. In this way, the model was able to capture critical aspects of the data including interesting phenomena such as waiting behavior. This indicates that in complex dynamic tasks participants may be aware of the consequences of their actions over relatively small time intervals.

This study contributes to our understanding about strategy use in complex dynamic tasks: which strategies are used, how they are selected, and how strategy execution changes as experience is gained. Good performance is linked to an effective combination of strategic control with attention to changing task demands.

The cognitive model also prescribes a mechanism in which environmental feedback controls how actions are selected in a highly dynamic task. Through the implementation of the cognitive model it was found, for example, that strategy execution depends on the fine-tuning of ACT-R production rule utilities as a consequence of

environmental rewards. Selecting actions based on utility comparisons facilitates a fluid and quick selection of actions that is instrumental in obtaining good performance, particularly in dynamic and time pressured situations. In dynamic tasks there is a continuous competition between top-down and bottom-up control. This competition is mediated by the characteristics of the learning process such as those exemplified in the Cañas et al. (2005) study, for which in the CT condition the top-down form of control dominates. The account provided by the model is that rules implementing top-down strategic control come to dominate behavior increasingly over rules implementing bottom-up responsive behavior during the CT phase owing to task consistency. This phenomenon increases the probability of performing well in the CT problem scenario but also produces cognitive inflexibility.

As mentioned in the introduction, Schunn & Reder (1996) found no evidence for cognitive inflexibility in their ATC study regarding strategy selection (choice of runway – long or short – on which to land aircraft) despite a long training period. However, we can learn from the work presented here; this would indicate that rules involved in the selection of choices in behaviour (for example, choosing between runways on which to land aircraft) have similar utility. A critical factor that enabled the dominance of certain rules in FireChief was the high consistency of the CT trial. In this respect, the ATC task is only partially consistent. An examination of the Ackerman (1988) study, from which the data for the second experiment of Schunn & Reder (1991) was extracted, reveals that weather conditions (wind speed, wind direction, and ground condition) varied randomly about twice a minute, and also that within each trial aircraft type, of which there are four, are randomly drawn from the queue. It seems that this experimental design shares more similarity with the VT condition in the Cañas et al. (2005) study rather than the CT condition, so when experimental changes are introduced no subset of rules has become dominant.

This research also provides an explanation of how dynamic tasks can be modeled using the Competing Strategies paradigm by incorporating an additional layer of within-strategy execution competition, enabling the bottom-up manifestation of strategies, such as that described here.

Acknowledgments

This research is funded by CONACYT. Special thanks to Hewlett-Packard GADSC MEX DEV Center for a conference grant.

References

Ackerman, P. L. (1988). Determinants of individual differences during skill acquisition: Cognitive abilities and information processing. *Journal of Experimental Psychology: General*, 117, 288-318.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review* 111, (4). 1036-1060.

Brehmer, B., & Dörner, D. B. (1993) Experiments With Computer-Simulated Microworlds: Escaping Both the Narrow Straits of the Laboratory and the Deep Blue Sea

of the Field Study. *Computers in Human Behavior*, Vol. 9, pp. 171-184.

Cañas, J.J., Antolí, A., Fajardo, I., Salmerón, L. (2005) Cognitive inflexibility and the development and use of strategies for solving complex dynamic problems: effects of different types of training. *Theoretical Issues in Ergonomic Science*, 6 (1) 95-108.

De Obeso Orendain, A., & Wood, S. (2010). Cognitive Modeling of Strategies in Dynamic Tasks. In D. D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the 10th International Conference on Cognitive Modeling* (pp. 52-57). Philadelphia, PA: Drexel University.

Gonzalez, C., Thomas, R., & Vanyukov, P. (2004). The relationships between cognitive ability and dynamic decision making. Elsevier *Intelligence* 33 (2005) 169–186.

Lovett, M. C. & Anderson, J. R. (1996). History of success and current context in problem solving: Combined influences on operator selection. *Cognitive Psychology*, 31(2), 168-217.

Omodei, M.M., Wearing, A.J. (1995). The FireChief microworld generating program: An illustration of computer simulated microworlds as an experimental paradigm for studying complex decision-making behavior. *Behav. Res. Meth. Instr. & Comp.*, 27, 303-316.

Peebles, D. & Bothell, D. (2004). Modelling performance in the sustained attention to response task. In *Proceedings of the sixth International Conference on Cognitive Modeling* (pp. 231-236). Pittsburgh, PA: Carnegie Mellon University/University of Pittsburgh.

Schoelles, M. J., & Gray, W. D. (2000). Argus Prime: Modeling emergent microstrategies in a complex simulated task environment. In N. Taatgen, & J. Aasman (Eds.), *Proceedings of the Third International Conference on Cognitive Modeling* (pp. 260-270). Veenendaal, NL: Universal Press.

Schunn, C. D., & Reder, L. M. (2001). Another source of individual differences: Strategy adaptivity to changing rates of success. *Journal of Experimental Psychology: General*, 130(1), 59-76.

Stewin, L., & Anderson, C. (1974). Cognitive complexity as a determinant of information processing. *Alberta Journal of Educational Research*, 20(3), 233-243.

Taatgen, N. A., Lebiere, C. & Anderson, J.R. (2006). Modeling Paradigms in ACT-R. In R. Sun (ed.), *Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation*. Cambridge University Press; 29-52.

ICCM Symposium on Cognitive Modeling of Processes “Beyond Rational”

William G. Kennedy (wkennedy.gmu.edu)

Krasnow Institute for Advanced Study, Center for Social Complexity
George Mason University, Fairfax, VA USA

Frank E. Ritter (frank.ritter@psu.edu)

The College of Information Sciences and Technology
The Pennsylvania State University, University Park, PA 16802 USA

Christian Lebiere (cl@cmu.edu), Ion Juvina (ijuvina@cmu.edu), Alessandro Oltramari (aoltrama@andrew.cmu.edu)

Department of Psychology
Carnegie Mellon University, Pittsburgh, PA 15213 USA

Jonathan Gratch (gratch@ict.usc.edu)

Institute for Creative Technologies and Department of Computer Science
University of Southern California, Playa Vista, CA USA

Richard M. Young (r.m.young@acm.org)

UCL Interaction Centre
University College London, UK

Abstract

Computational cognitive modeling is normally thought of as rational cognition. However, there are human behaviors that do not appear to be driven by rational cognition. The other, “beyond rational” cognition is also appropriate for computational models of cognition. The panel will discuss their efforts at modeling this form of cognition.

Keywords: cognitive models; cognition; Dual Process Theory, emotion, intuition.

Introduction

Cognitive modeling has been primarily aimed at implementing and testing theories explaining behavior driven by rational, multi-step cognition and it has been very successful (e.g., Anderson, 2007; Anderson, et al., 2004; Laird, 2008; 2012). However, there are many human behaviors that seem to be driven by aspects of behavior that are not the same as “rational” cognition: immediate judgments, intuitive, emotional, and other non-rational, hence “beyond rational” processes. These aspects may result in such phenomena as emotional natural language generation, optical illusions, snap judgments, and humor.

There has been a growing literature on these processes. Significant books include LeDoux’s *The Emotional Brain*, Gigerenzer’s *Gut Feelings*, Klein’s *Sources of Power*, Thagard’s *Hot Thought*, Minsky’s *The Emotion Machine*, and Irvine’s *On Desire*. Herbert Simon also addressed this topic in his *Reason in Human Affairs*. However, computational models of non-rational cognition are relatively rare (cf., Gratch & Marsella, 2004; Kennedy & Bugajska, 2010).

The Dual Process Theory could provide a basis for computational cognitive modeling of these aspects. The

Dual Process Theory suggests two types of processes drive behavior (Evans, 2008; Sloman, 1996). Reality may be more nuanced than a simple dichotomy and this grouping is somewhat controversial. The more neutral terms for the two processes are System 1 and System 2, with System 2 being the rational, conscious, multi-step, slower, more evolutionarily advanced process (Kahneman, 2003). The implicit learning discussion of a few years ago could provide examples of one or the other side, rather than trying to fit all implicit learning phenomena within one side (Wallach & Lebiere, 2002). It may also be that rather than two processes, there may be a spectrum of processes between two extremes or cognition may have more dimensions than one. There is a suggestion that much of our behavior is the result of this other reasoning.

This panel will address the topics related to cognitive modeling of beyond-rational cognition. The panel members will present their views on the topic and whether it would be appropriate for the cognitive modeling community to entertain models of behavior driven by beyond rational processes.

Panel Makeup

The panel consists of cognitive modelers who have thought about this topic. Each has provided an abstract of their input to this topic.

William G. Kennedy

Starting with the ancient Greeks, we have believed that there were two forms of cognition that control our behavior: passion and reason, and that there was an inner battle for control of the mind (LeDoux, 1996). Dualism, proposed by Descartes, separated mind and body and has been

discredited in current philosophy (Evans, 2010). When we began to study cognition scientifically, William James considered reasoning, consciousness, emotion, instinct, and will as separate topics, although consciousness received the shortest treatment (James, 1892/2001).

With the cognitive revolution of the second half of the 20th century came a focus on testable theories of Cognitive Science and the verbal descriptions of non-rational cognition have been marginalized. However, recently there has been resurgence in interest in the other side, the intuitive, emotional side of cognition. There have been many books written on how people make decisions using methods outside traditional rational cognition. Dualism has evolved through a dual representation of knowledge, visual and verbal (Paivio, 1971) into a Dual Process theory of cognition (Evans, 2008; Sloman, 1996).

The Dual Process theory suggests a distinct separation of cognitive processes and they can be organized into (at least) four groupings: consciousness, evolution, functional characteristics, and individual differences (Evans, 2008). For example, Table 1 presents the functional characteristics of the two systems.

Table 1: Functional Characteristics of the Dual Process Theory (from Evans 2008).

System 1	System 2
Associative	Rule-based
Domain specific	Domain general
Contextualized	Abstract
Parallel	Sequential

In addition to the concept and the separation of characteristics, even the naming of the two systems is controversial and calling them System 1 and System 2 is an attempt to keep the discussion focused on the content, not the naming (Gray, 2007; Kahneman, 2003).

As an indication of the trend in the interest in the topic, Figure 1 is offered. The figure shows the frequency of searches for the term “System 1 System 2”

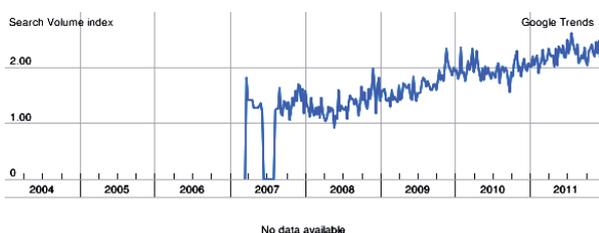


Figure 1. Google Trending of “System 1 System 2”

With the general attention to the concept of dual processes of cognition, the cognitive modeling community should consider more seriously modeling cognition “beyond rational.”

Frank Ritter

Frank will discuss work on modeling the effects of caffeine on behavior, and work on modeling the effects of stress on behavior. These approaches have been done with sets of changes overlaying the cognitive architecture. More recent work suggests that perhaps this approach is productive in the short term, but that a longer term solution is to model the physiological substrate that cognition is based upon (Dancy, Ritter, & Berry, 2012; Ritter, Dancy, & Berry, 2011), as well as modeling more complex cognition including multiple types of appraisal and process monitoring.

Christian Lebiere/Ion Juvina/Alessandro Oltramari

Cognitive architectures such as ACT-R (Anderson & Lebiere, 1998; Anderson et al., 2004) have been quite successful at formalizing and organizing basic cognitive processes in computational frameworks that can accomplish complex tasks. Contrary to common descriptions as purely symbolic or rational, they actually integrate both explicit and implicit cognitive processes, including declarative and procedural knowledge as well as symbolic and subsymbolic levels of representation. The duality of System 1 (automatic) vs. System 2 (controlled) processes is thus an oversimplification of the reality of complex cognition, which integrates basic, intuitive steps of cognition driven by the subsymbolic parameters of the knowledge structures involved into controlled threads of execution capable of accomplishing complex tasks. Despite the success of this purely cognitive approach at modeling a broad range of cognitive tasks, we have found it necessary to contemplate integrating emotional processes into the architectural framework.

The mainstream approach (e.g., Gratch, Marsella et al., 2009; Marinier, Laird, et al., 2009) is concerned with modeling discrete emotions as they arise from appraisal processes that are hardwired in the architecture. Our approach to modeling affective processes is complementary to that approach. We claim that only psychological primitives need to be included in the architecture. Psychological primitives are basic mechanisms that allow us to learn and adapt to the environment. Perceptual experiences, knowledge and skills are not to be included in the architecture. They can be part of specific models and are usually developed through various learning mechanisms. According to this view, discrete emotions are not psychological primitives. They are not biologically given (Barrett, 2006) but instead develop (are learned) from core affect. We conceive of emotion as a perceptual-conceptual experience that is analogous to color perception. People use category knowledge about color to shape the perception of wavelengths of light into the experience of color (Barrett, 2006). Correspondingly, people use category knowledge about emotion to shape the interoception of core affect into the experience of emotion. Core affect is the constant stream of transient alterations in an organism’s neurophysiological

state that represent its immediate relation to the flow of changing events (Russell, 2003). It is typically characterized along two (or three) dimensions: valence and arousal (and approach-avoidance). Changes in core affect can result from physiological (e.g., hunger) and cognitive processes (valuation). Valuation is the process of learning the (expected) value of stimuli encountered in the environment. Very few stimuli have intrinsic value (i.e., they act directly on our nervous system without involving prior learning). Typically, people learn the value of stimuli by associating them with core affect states and external events.

We have developed a simple valuation mechanism that associates a specific value to every representation (chunk). These values are called valuations and can be used to evaluate new stimuli. They are learned via a reinforcement learning mechanism similar to the mechanism of learning the utilities of actions. Thus, the valuation of a chunk is a learned expectation of the likelihood that the chunk would be relevant to the current situation. The relevance indicated by valuation is additive to that indicated by activation. The sign and magnitude of valuation can be used as constraints on retrieval. Valuations are computed based on the rewards that the model receives during its execution and they change as the model is executed.

We claim that activation and valuation (together with learning) are the necessary and sufficient architectural building blocks of cognitive and affective processing. We are using these mechanisms to develop specific models in which cognition and affect interact to produce human-like goal-directed adaptive behavior. For example, in a variant of the game Prisoner's Dilemma, we showed that a cognitive model was more effective than the human participants. Specifically, it learned that cooperation was more beneficial in the long term, and it did not react to occasional unreciprocated attempts to cooperate (Juvina, Lebiere, et al., 2011). However, human participants showed signs of emotional reactivity. Particularly, they were more likely to immediately react by defecting after unreciprocated cooperation, ignoring the potential long-term benefits of sustained cooperation. This behavior has been observed in other studies with similar tasks and associated with a specific pattern of neural activity (e.g., Rilling, 2008). In order to correct for the mismatch between model and human data, we introduced an emotional bias in the model. The assumption was that such a bias develops in human-human interactions to prevent exploitation of a player by another. We claim that such emotional biases are learned from interaction experience using the architectural mechanism described above.

Jonathan Gratch

As someone that studies and models emotion, I definitely agree there is value in a symposium on "beyond rational" processes, but I will take issue with the perspective that attempts to dichotomize cognition and characterizes traditional/successful cognitive modeling as sequential and deliberative. In general, I have a problem with dual process

explanations which (in my view) tend to overly simplify cognition as either: emotional vs. rational; intuitive vs. deliberative; or "System 1" vs. "System 2. Rather, I will argue that dual-process distinctions are largely an artifact of how we study and formalize cognition. On the one hand, normative frameworks for formalizing cognition (e.g., decision theory, game theory or Bayesian inference) highlight human departures from "rational behavior" that may say more about the limits of our frameworks than the duality of human cognitive processes (e.g., see Gigerenzer, 1991). On the other hand, experimental paradigms that illustrate such dualities present participants with unnatural situations designed to highlight these distinctions. Instead, I see thought arising from a tight coupling and dynamic unfolding of a variety of processes (some more naturally characterized as automatic/parallel and some more naturally characterized as sequential).

I am also not convinced that cognitive models are most naturally seen as simply sequential/deliberate. Even early cognitive architectures such as Soar (Newell, 1990) have this close coupling of "automatic" (e.g., elaborations) and deliberative/sequential (e.g., operators) processes (although we might quibble about if this maps well onto any specific dichotomy), and many "successful" cognitive models (e.g. Thagard's 2002 coherence models); models of perceptual or motor processes) are not naturally viewed as sequential.

Despite these quibbles about dual process models, I fully agree that cognitive science, and especially the cognitive modeling community, have largely ignored modeling problems that involve emotion and motivation with the consequence that, on the one hand, we are sorely lacking when it comes to information processing accounts of emotional processes. On the other hand, cognitive models tend to overlook a whole class of problems and mechanisms that might give a different window on how cognition works outside the emotionally-sheltered laboratory.

Richard Young (Discussant)

Richard Young has a long-standing interest in cognitive modeling, cognitive architectures, and related matters. He will respond to the presentations in the symposium, doing his best to identify common threads and contentious themes, before opening the discussion to the audience.

Acknowledgments

This work of the first author was supported in part by AFOSR/AFRL grant FA9550-10-1-0385, the Center of Excellence in Neuroergonomics, Technology, and Cognition (CENTEC), by the Center for Social Complexity at George Mason University, and by the Office of Naval Research (ONR) under a Multidisciplinary University Research Initiative (MURI) grant no. N00014-08-1-0921. The work presented by Frank Ritter was supported by ONR (N00014-03-1-0248) and the Defense Threat Reduction Agency (DTRA). Christian Lebiere (joint work with Ion Juvina and Alessandro Oltramari) is supported by DTRA grant number: HDTRA1-09-1-0053 to Christian Lebiere. Jonathan Gratch

is supported by AFOSR grant FA9550-09-1-0507 and the National Science Foundation grant IIS-0916858.

References

- Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* Oxford: Oxford University Press.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglas, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of mind. *Psychological Review* 111(4): 1036-1060.
- Dancy, C. L., Ritter, F. E., & Berry, K. (2012). Towards adding a physiological substrate to ACT-R. In *Proceedings of the 21th Conference on Behavior Representation in Modeling and Simulation*. BRIMS Society: Amelia Island, FL.
- Feldman Barrett, L. (2006) Are Emotions Natural Kinds? *Perspectives on Psychological Science*, 1, 1, 28-58.
- Evans, J.St.B.T. (2008). Dual-processing accounts of reasoning, judgment and social cognition. *Annual Review of Psychology*, 59, 255-278.
- Gigerenzer, G. (1991). How to Make Cognitive Illusions Disappear: Beyond "Heuristics and Biases". *European Review of Social Psychology* 2(1), 83-115.
- Gigerenzer, G. (2007). *Gut Feelings: The Intelligence of the Unconscious*. New York, NY: Penguin Books.
- Gratch, J. & Marsella, S. (2004) A Domain-independent Framework for Modeling Emotion. *Journal of Cognitive Systems Research*, 5(4), 269-306.
- Gratch, J., Marsella, S., Wang, N., & Stanokovic, B. (2009). Assessing the validity of appraisal-based models of emotion. *Proceedings of the International Conference on Affective Computing and Intelligent Interaction*.
- Gray, W. D. (2007). Composition and control of integrated cognitive systems. In W. D. Gray (Ed.), *Integrated models of cognitive systems*, 3-12. New York, NY: Oxford University Press.
- Irving, W.B. (2006). *On Desire: Why we want what we want*. Oxford: Oxford University Press.
- James, W. (1892/2001) *Psychology: The Briefer Course*. Dover edition republication of Harper Torchbook edition 1961 abridged of original work first published by Henry Holt and Co. 1892. Toronto, Ontario: Dover edition.
- Juvina, I., Lebiere, C., Martin, J.M., & Gonzalez, C. (2011) Intergroup Prisoner's Dilemma with Intragroup Power Dynamics. *Games*, 2, 21-51.
- Kahneman D. (2003). A perspective on judgment and choice. *American Psychologist*. 58, 697-720.
- Klein, G. (1998). *Sources of power: How people make decisions*. Cambridge, MA: The MIT Press.
- Kennedy, W.G. & Bugajska, M. (2010). Integrating Fast and Slow Cognitive Processes. In D.D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the International Conference on Cognitive Modeling (ICCM 2010)* (pp. 121-126). Philadelphia, PA, (4-8 August 2010)
- Laird, J.E. (2008). Extending the Soar Cognitive Architecture. *Proceedings of the Conference on Artificial General Intelligence*. Memphis, TN: ISO Press.
- Laird, J.E. (2012 in press). *The Soar Cognitive Architecture*. Cambridge, MA: The MIT Press.
- LeDoux, J.E. (1996). *The Emotional Brain: The Mysterious Underpinnings of Emotional Life*. Simon & Schuster.
- Marinier, R.P., Laird, J.E., & Lewis, R.L. (2009) A computational unification of cognitive behavior and emotion. *Cognitive Systems Research*, 10, 48-69.
- Minsky, M. (2006). *The Emotion Machine: Commonsense Thinking, Artificial Intelligence, and the Future of the Human Mind*. New York, NY: Simon & Schuster.
- Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.
- Paivio, A. (1971) *Imagery and verbal processes*. New York, NY: Hold, Rinehart, & Winston.
- Rilling, J.K. (2008) Neuroscientific Approaches and Applications within Anthropology. *Yearbook of Physical Anthropology*, 51, 2-32.
- Ritter, F. E., Dancy, C., & Berry, K. (2011). The case for including physiology in cognitive modeling. In *Biologically Inspired Cognitive Architectures*, 303. Amsterdam: IOS Press.
- Russell, J.A. (2003) Core Affect and the Psychological Construction of Emotion. *Psychological Review*, 110, 1, 145-172.
- Simon, H.A. (1982). *Reason in Human Affairs*. Stanford University Press.
- Slooman, S.A. (1996). The empirical case for two systems of reasoning. *Psychological Bulletin*, 119, 3-22.
- Thagard, P. (2002). *Coherence in Thought and Action (Life and Mind: Philosophical Issues in Biology and Psychology)*. Cambridge, MA: The MIT Press.
- Thagard, P. (2006). *Hot Thought: Mechanisms and Applications of Emotional Cognition*. Cambridge, MA: The MIT Press.
- Wallach, D. & Lebiere, C. (2003). Implicit and explicit learning in a unified architecture of cognition. In L. Jimenez (Ed.) *Advances in Consciousness Research*, 215-250. Amsterdam: John Benjamins Publishing Company.

COMPUTATIONAL SIMULATION OF VISUAL DISTRACTION EFFECTS ON CAR DRIVERS' SITUATION AWARENESS

Thierry Bellet¹, Jean-Charles Bornard^{1,3}, Pierre Mayenobe¹, Jean-Christophe Paris^{1,3},
Dominique Gruyer², Bernard Claverie³

¹IFSTTAR (LESCOT), 25 Avenue François Mitterrand, 69675 Bron cedex, France

²IFSTTAR (LIVIC), 14 Route de la minière, 78000 Versailles, France

³ENSC (CIH), 146, rue Léo Saignat, 33076 Bordeaux, France

(corresponding author : thierry.bellet@ifsttar.fr)

Abstract

This paper presents a computational modeling approach for negative effects simulation of visual distraction while driving a car. In order to investigate these effects, an experiment was firstly implemented on a driving simulator. Twenty participants were invited to perform a car following task in different driving conditions (12 driving scenarios), with or without a secondary task of visual distraction. Empirical data collected through this experiment show that visual distraction negatively impacts the driving performance at both perceptive and behavioral levels, and then increase the risk of having a crash. Beyond these effects on the observable performance, the aim of this study is also to investigate and simulate these distractive effects on mental models of the road environment. Indeed, driver's decisions and behaviors are based on a temporal-spatial mental model, corresponding to the driver's *situation awareness (SA)*. This mental representation must be permanently updated by perceptive information extracted from the road scene to be efficient. In case of visual distraction requiring off-road scanning, mental model updating is imperfectly done and driver's actions are thus based on a mental representation that can dramatically differ from the situational reality, in case of a critical change in the traffic conditions (e.g. sudden braking of the lead car). From these empirical results, a computational model (named COSMODRIVE for *Cognitive Simulation Model of the DRIVER*) was implemented for simulating visual distraction effects and human errors risks at perceptive (visual scanning changes) cognitive (erroneous Situation Awareness) and behavioral levels (late reaction time and crash risk increasing).

Keywords: Computational Model, car Driver, Visual Distraction, Situation Awareness, Temporal-Spatial Mental Representation.

1. Introduction: Visual distraction and research objective in terms of computational simulation

Driving requires visual attention in order to safely control the car and to respond to events happening on the road. Driver distraction occurs "*when a driver is delayed in the recognition of information needed to safely accomplish the driving task because some event, activity, object, or person within or outside the vehicle compelled or tended to induce the driver's shifting attention away from the driving task*" (Treat, 1980). In the same way, distraction has been more recently defined by Lee, Regan and Young (2009) "*as a diversion of attention away from activities critical for safe driving toward a competing activity*".

Two main forms of distraction are commonly described in the literature, namely visual and cognitive distraction. The former takes a driver's "eyes-off-road", while the latter takes their "minds-off-road" (Victor et al., 2005). The present study is focusing only on visual distraction due to a secondary visual task taking the driver's eyes off the road.

This type of distraction can occur when drivers look at in-vehicle displays. For example, in research conducted by Wierwille et al. (1988) under real traffic conditions where text was displayed on an on-board screen, the average length of a glance at the outside environment was 1.5 to 1.7 seconds, while the amount of time spent watching the road decreased to about 50 to 65% of total eye movement. Visual scanning toward on-board-devices varies with the nature of displayed information and the type of additional task to be performed while driving, but also according to the situational demand and driver's adaptation strategies regarding both the driving situation and the demand and the reading task demands. However, focusing visual attention for some period of time on in-vehicle visual target creates an unsafe driving issue. Senders et al. (1967) argued that when drivers look away from the road, uncertainty about the roadway situation increases. When uncertainty reaches a certain threshold, drivers look back to the road. More recently, Wierwille (1993) quantified this threshold of off-road glance duration at 1.8 seconds on a straight road and 1.2 seconds on a curve on average for a normal driver. Such thresholds may also vary according to driver speed or traffic condition and may be also subject to individual differences.

Changes in driver behavior due to visual distraction have been identified in simulator or in vivo studies. Several studies have shown that visual distraction increases the dispersion of eye gaze pattern from the roadway (e.g. Donmez et al., 2007). In terms of driving performance, visual distraction has been also associated in the literature with large, discrete steering adjustments and increased lane deviations (e.g. Engström, Johansson and Östlund, 2005; Salvucci, 2001). However, as discussed by Zhang (2011), such inference has been mainly assessed for lower-level of driving control and less is known concerning the internal cognitive effect of visual distraction. Moreover, as explained this author, "*a data-driven approach to identifying detrimental effects of distraction may not be sufficient to establish a causal link between driver performance and visual process interference. Modeling internal changes in driver Situation Awareness of the driving environment due to visual distraction are required to conclusively identify precise relationships between drivers' performances and distraction and support effective mitigation strategies for distractions*". This is typically what this research would like to explore. Beyond the well-known impact of visual distractions on drivers' visual strategies and driving performance at the operational level (e.g. Salvucci, 2001), the aim of this research is above all to

develop a computational model able to simulate these distractive effects on drivers' Situation Awareness modeling as a dynamic visual-spatial model of the surrounding.

2. Empirical data collection among human drivers to study visual distraction effects

The methodological specificity of the driver modelling approach implemented in this research was to use the same virtual Platform (named SIVIC; Gruyer et al., 2006) as (i) a driving simulator for empirical data collection among human drivers, and then, as (ii) a virtual road environment to be interfaced with the driver model for virtual simulations (in charge to reproduce humans' performances). According to this approach, human drivers' behaviour and driver model performances were observed and simulated for the same driving scenarios, in the same virtual road environment.

2.1 Apparatus

The experiment used a fixed-base simulator integrating a real car seat, three PC monitors for presenting the driving scene (the back mirror view is computationally integrated in the central image), and a *Logitech G 25 kit* including the steering wheels, 3 pedals, a gear box, and indicators. Two web-cameras were used for recording drivers' face and feet movement on the pedals. A third video camera was also added behind the car seat, in order to film the driving environment and the driver's activity. A 12-inch tablet computer was placed in front of the main simulator screens. This display was used to present the visual distraction tasks to the drivers. This screen was positioned approximately 15 degrees down and 30 degree right of the natural line of sight of participants in viewing the driving scene.

2.2 Participants

Twenty experienced drivers of middle-age (from 23 to 56 years old) participated to this experiment. All the drivers have a minimum of 5 years of driving experience and they drive a minimum of 5.000 km per year. The recruitment of subjects was balanced for gender. Participants were instructed to perform the secondary task in accordance with the demands of the driving situation. The instruction emphasized that safe driving was of the highest priority.

2.3 Driving task

The full experiment followed a 3×2×2 factorial design with one primary driving task of car following to be performed in three different driving contexts (requiring different driving speeds: 130 km/h for Highway, 90 km/h for rural roads and 50 km/h for urban areas), from two required following distances (free *versus* imposed at a value of 0.6 second of Inter-Vehicular Time [IVT]), and two types of lead car behavior (having a steady *versus* irregular velocity) and then, two levels of visual distraction (with and without). In total, there were 12 driving scenarios to which each participant was exposed, once time without any secondary task, and then, on time with a secondary task.

Each scenario was around 1 minute in duration and presented one experimental condition.

2.3 Visual secondary task

The Secondary Task of visual distraction to be performed by the participants was the following: a set of 3 visual pictograms, associated with an auditory beep, were displayed on an additional screen situated on the right side (near the usual position of the radio). Some seconds later (from 3 to 4 sec.), 1 of this 3 pictograms appeared under the first set, and the driver had to use a 3-buttons command for indicating which pictogram is replicated (Fig. 1).

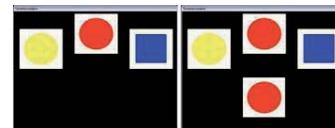


Figure 1: The visual Secondary Task to be performed

2.4 Main results

2.4.1 Visual strategies for additional screen scanning

Visual strategies during secondary task have been extracted from the analysis of video film of participants' faces collected during the experiment. The two main different visual scanning patterns of the additional-screen observed among human drivers are presented in figure 2 (others strategies are adaptations of one of the two main patterns).

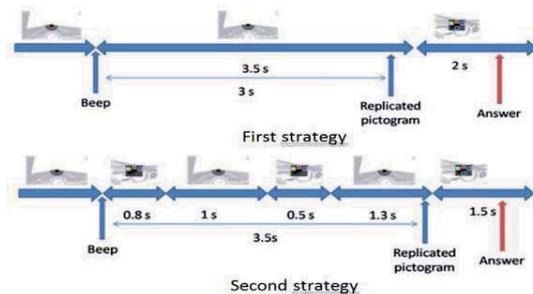


Figure 2: Visual scanning patterns observed among human drivers

The first strategy (58 % of the cases, and more systematically used by 40% of the participants) consists in waiting from 3 to 4 seconds when the beep occurs, hoping that only one screen scanning will allow the participant (a) to see the 3 pictograms, (b) to see the replicated one, and (c) to provide the answer. The main advantage of this strategy is to reduce the number of glance, but the convenient is to require a long glance of 2 seconds (in mean), for processing all the pictograms. The second strategy (observed in 31 % of the cases, and more systematically used by 25% of the participants), was to look at the screen briefly when the beep occurs (mean duration of 0.8 sec.), in order to observe the 3 pictograms, and then to go back to the road scene while regularly checking the screen (via brief glances of 0.5 sec) until the replicated pictogram appears. When it occurs, a more long off-road glance (around 1.5 sec.) is implemented for checking the replicated pictogram and validating the

answer. By contrast with the preceding one, this second strategy requires a several glances, but the advantage is to process visual information in two times (corresponding to *question*, and then *answer*), requiring a shortest last glance.

2.4.2 Visual distraction effect on driving performance

Two main negative impacts of a visual distraction on the drivers' performances were observed during this experiment. The first one occurs in normal conditions, and the second one occurs for critical scenarios (i.e. when the lead car brakes), increasing the accident risk.

In *normal driving conditions*, two main differences due to visual distraction were observed among the participants: (i) a significant reduction (T-test, $p < 0.001$) of the safety margins in *free following conditions* (without ST, mean value of IVT is of 3 s. *without ST*, vs 2.65 s. with ST) and (ii) a significant degradation ($p < 0.05$) of the following performance in constrained following conditions (in these scenarios, drivers have to follow the lead car at an imposed IVT of 0.6 s., and the percentage of time when this value is performed is of 57% without ST, vs 44 % with ST). These results show a negative effect of visual distraction for short following distance keeping.

In *critical driving conditions*, the two main negative impacts of the visual ST on drivers' performances are (i) an increasing of reaction time for braking (the differences are only significant for the constrained following task : 0.89 s. vs 1.1 s.; $p < 0.05$), and (ii) a risk of crash increasing. The Table 1 presents the percentages of collision occurring with the lead car for the total number of required emergency braking, by respectively considering the different driving scenarios investigated. It appears that the risk of collision due to a visual distraction is here significantly increased for 4 of the 10 driving scenarios requiring an emergency braking (i.e. bold values). The highest negative impacts of visual ST were observed for the *constrained unsteady car following* scenarios, in both urban and rural environments.

Table 1: Percentages of collision with the lead car

Context	Driving scenario	No ST	With ST
Highway	Free steady lead car following	55 %	50 %
	Free unsteady lead car following*	35 %	50 %
	Constrained steady lead car following	65 %	70 %
	Constrained unsteady lead car following	70 %	70 %
Rural	Free unsteady lead car following	60 %	60 %
	Constrained unsteady lead car following*	55 %	80 %
Urban	Free steady car lead following*	20 %	30 %
	Free unsteady lead car following	30 %	30 %
	Constrained steady lead car following	30 %	30 %
	Constrained unsteady lead car following*	25 %	90 %

(*Bold Values indicate the main observed differences in driving performance due to visual distraction)

2.4.3 Example of crash due to visual distraction

The following figure presents a typical case of driving accident due to visual distraction, as observed during this experiment (in free following conditions, *view a*). In this example, the lead car brakes when the driver is looking for the additional screen (*view b*), via a long glance of 2 seconds. When she repays attention to the road (*view c*), she however discovers a critical gap between the expected position of the lead car as mentally assessed during the off-road glance (by assuming a steady speed of the lead car during this period) and the objective reality where the lead car is actually very close. Therefore, she immediately carried out an emergency braking (0.78 second of reaction time). Unfortunately, the collision cannot be avoided, and the crash with the lead car occurs in *view d*.

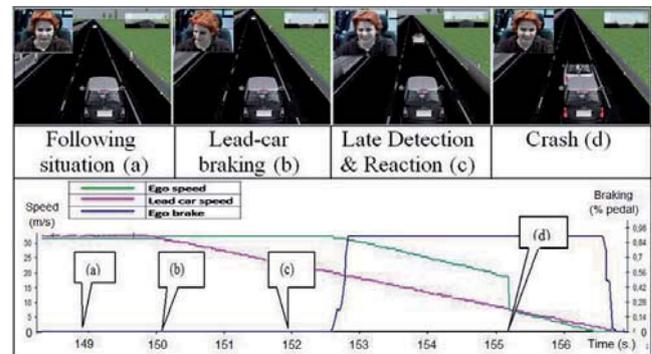


Figure 3: Typical example of crash due to visual distraction

3. Computational modeling and simulation of visual distraction effects on drivers' SA

By using the empirical data collected in this experiment, a computational model, based on the COSMODRIVE (*COgnitive Simulation MODEL of the DRIVER*) theoretical approach (Bellet et al., 2007), has been implemented into the SIVIC virtual plate-form (Gruyer et al, 2006). By contrast with other driver models available in the literature, the core specificity of COSMODRIVE is to simulate drivers' mental representation as a visual-spatial (i.e. 3 Dimensional) and dynamic model of the road environment. Indeed, from their interaction with the road environment, drivers build mental model of events and objects surrounding them. This mental model corresponds to the driver's *Situation Awareness* (Endsley, 1995). They are dynamically formulated in working memory through a matching process between perceived information and pre-existing *operative* knowledge (Ochanine, 1977). At the *tactical level* (Michon, 1985), such a mental representations provides an ego-centred and a goal-oriented understanding of the traffic situation. They take the form of a dynamic 3D model of the road environment, liable to be mentally explored by the driver in order to anticipate events or action effects through cognitive simulations of *mental deployment* (Bellet et al, 2010; 2011), and thus providing expectations on future situational states. This cognitive process of anticipation, based on both *implicit* and *explicit* mental

simulations (Bellet et al., 2009), is a core function of the human cognitive system in dynamic contexts. The central structure supporting to the driver's SA in COSMODRIVE cognitive architecture is *working memory*. From this point of view, this architecture is inspired by the ACT-R theory (Anderson et al., 2006). However, the working memory of COSMODRIVE merges both *procedural* and *declarative* memories, and comes more from the *operational memory* concept of Zintchenko (1966) than from the Baddeley's *working memory* model (1986). With COMSODRIVE, car driving is modeling as a dynamic regulation loop of interaction between drivers and the road environment.

Figure 4 provides a synthetic overview of this model as implemented on the SIVIC virtual platform. Synthetically the functional architecture of the model is based on 3 main modules (i.e. Perception, Cognition, and Action modules), in order to drive a virtual car into a virtual environment, through two synchronized "Perception-Cognition-Action" regulation loops: an automatic and implicit mode versus an attentional and explicit mode (Bellet et al, 2009). This dichotomy is well established in scientific literature, for example, with the distinction put forward by Schneider and Schiffrin (1977) between controlled processes, which require cognitive resources and which can only be performed sequentially, and automatic processes, which can be performed in parallel without any attentional effort. In the same way, Rasmussen (1986) distinguishes different levels of activity control according to whether the behaviours implemented rely on (i) highly integrated sensorial-motor reflexes (Skill-based behaviors), (ii) well mastered decision rules for managing familiar situations (Rule-based behaviors), or (iii) more generic knowledge that is activated in new situations, for which the driver have not any prior experience (Knowledge-based behaviors).

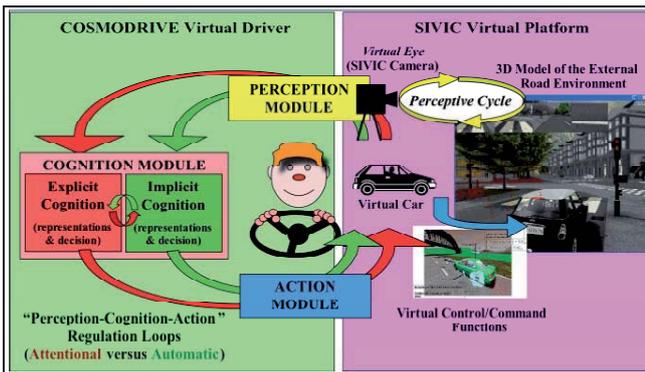


Fig. 4: Architecture of COSMODRIVE model

From this architecture, the *Perception Module* is in charge to simulate human information processing, the *Cognition Module* is in charge to simulate mental representation elaboration (SA) and decision-making processes at both the *attentional* and *automatic* levels, and the *Action Module* is in charge to simulate executive functions and vehicle control abilities, allowing the model to dynamically progress on the SIVIC virtual road by driving a virtual car.

3.1 The Perception module

The Perception Module acts as an "interface" between the external road environment (as simulated with SiVIC) and the driver model. It simulates human information processing of sensorial data before their integration in the Cognition module for traffic conditions analysis, situational change anticipation, decision making, and then action planning and implementation through the Action Module. The *Perception module* is based on a *virtual eye* (Figure 5). This virtual eye includes three visual field zones: the central zone corresponding to *foveal vision* (solid angle of 2.5° centred on the fixation point) with a high visual acuity, *para-foveal* vision (from 2.5° to 9°), and *peripheral* vision (from 9° to 150°), allowing only the perception of dynamic events.



Fig. 5: COSMODRIVE virtual eye

From this virtual eye, COSMODRIVE is able to integrate information perceived in the road environment through two main processes (Bornard et al, 2011). The first one, named *perceptive integration*, is a data-driven process (i.e. bottom-up) and allows the cognitive integration of environmental information in the driver's mental representations. The second one, named *perceptive exploration* and based on Neisser's *perceptual cycle theory* (1976), is a "knowledge-driven" process (i.e. top-down) in charge to actively explore the road scene, according to the tactical goal to be reached and to the event expectations included in the driving schemas. In the frame of a car following task, the main point of interest of the driver's visual attention is the lead car. However, in case of a visual secondary task to be performed while driving, the virtual eye must sometimes leave the road in order to observe the additional screen, according to the 2 different visual scanning patterns observed among human drivers during our experiment (presented in fig. 2).

3.2 The Cognition module

The Cognition Module is in charge to support drivers' Situation Awareness, Decision-Making and Action Planning. The specificity of COSMODRIVE architecture, by contrast with other driver models developed with ACT-R (e.g. Salvucci, 2006), is to be able to support dynamic reasoning based on visual-spatial mental models. This type of 3-Dimensional (3D) mental representation is a key component of real drivers' cognition (Bellet et al, 2009), but they are not easy to implement, and then to process, with the ACT-R architecture (except as a *chunk*, i.e. a set of facts or logical units, stored in *declarative memory*). In order to simulate human drivers' visual-spatial knowledge and dynamic reasoning in a more realistic way, we therefore defined a specific computational formalism named *driving schemas* (Bellet et al, 2007). Coming from both Piaget's

(1936) concept of *operative scheme* and the Minsky (1975) *frames theory*, a *driving schema* is a functional 3D-model of the road *Infrastructure* associated with a *Tactical Goal* to be reached in this infrastructure. It is made of a *Driving Path*, defined as a sequence of *Driving Zones*, and integrates a sequence of *Actions* to be progressively implemented when progressing on the path. The decision to implement or not an action depends of *Conditions* to be checked by the driver regarding the occurrence of *Events* in particular *Perceptive Zones* of the road infrastructure. An event is an *Object* with specific *Characteristics* (its aspect, behaviour, or status). Once activated in working memory and instantiated with the characteristics of the current road environment, the active driving schema becomes the *tactical mental representation* of the driver, that is continuously updated as and when s/he progresses on the road. It corresponds to the driver's situation awareness of the situation. In the frame of a car-following task on straight line as investigated in this paper, the driving schema is focused on the tactical goal of *progressing along the same road lane* (no overtaking), at a given speed, and keeping a safe distance with the lead car.

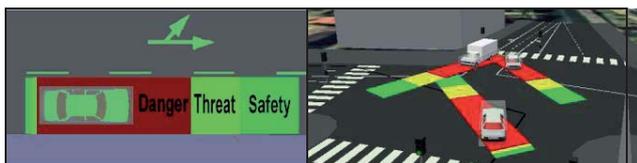


Figure 6: COSMODRIVE "Envelope-Zones" model

At the operational level, corresponding to an automatic control loop, COSMODRIVE regulation strategy is jointly based on *envelope zones* and *pure pursuit point* approaches. From a theoretical point of view, the concept of envelope zones comes from two classical theories in psychology: the notion of *body image* of Schilder (1950), and the theory of *proxemics* defined by Hall (1966), relating to the distance keeping in social interactions with other humans. Regarding car-driving activity, envelope zones also refer to *safety margins*. At this last level, COSMODRIVE model (Fig.6) is based on Konratos' work (1974) distinguishing a *safety zone*, a *threat zone*, and a *danger zone*. Envelope zones correspond to the portion of the path of driving schema to be occupied by the vehicle in the near future. As an "hidden dimension" of the social cognition, as suggested by Hall's theory (1966), these proxemics zones are also mentally projected to other road users, and are then used to dynamically interact with them, as well as to anticipate and manage the collision risks. This "virtual skin" is permanently active while driving, as an implicit awareness of our expected allocated space for moving. As with the Schilder's body schema, it belongs to a highly integrated cognitive level (i.e. implicit regulation loop), but at the same time, it favors the emergence of critical events in the driver's explicit awareness. Therefore, the envelope zones play a central role in the regulation of "social" as well as "physical" interactions with other road users under normal driving conditions (e.g. inter-vehicle distance keeping), and in the risk assessment of path conflicts and their

management, if a critical situation occurs (commitment of emergency reactions).

Moreover, two Decision-Making processes are implemented in COSMODRIVE model, one for each regulation loops presented in fig. 4. At the attentional level, corresponding to *explicit* decisions, this process is modelling through State-Transition automats intimately linked with the *driving path* and *conditions* integrated in tactical *driving schemas*. In real driving conditions, this tactical level is typically used for overtaking decision-making. However, in the frame of the empirical data collected in our experiment, primarily involving automatic driving abilities, the tactical level is mainly active when the lead car suddenly brakes and when the situation becomes critical. At the automatic level, an *implicit* decision-making is implemented through envelope zones, in order to keep a safety distance with the lead car (i.e. keep it in the green zone).

3.3 The Action module

The Action Module is in charge to perform vehicle-control skills, according to the driving actions decided and planned at the representational level by the Cognition module. The two core regulation mechanisms effectively implemented by the Action Module are based on (i) the *Pure-Pursuit Point method* and (ii) safety margin keeping by using *Envelope-Zones*. The *Pure Pursuit Point* method is used by COSMODRIVE for the lateral and the longitudinal controls of the car along the driving path of a tactical schema (Mayenobe, 2004). Mathematically, the pure-pursuit point is defined as the intersection of the desired vehicle path and a circle of radius centered at the vehicle's rear axle midpoint (assuming front wheel steer). Intuitively, this point describes the steering curvature that would bring the vehicle to the desired lateral offset after traveling a distance of approximately l . Thus the position of the pure-pursuit point maps directly onto a recommended steering curvature: $k = -2x/l$, where k is the curvature (reciprocal of steering radius), x is the relative lateral offset to the pure-pursuit point in vehicle coordinates, and l is a parameter known as the look-ahead distance. According to this definition, the operational control of the car by COSMODRIVE is a monitoring loop in charge to permanently keep the Pursuit Point in the driving path, to a given speed assigned with each segment of the tactical schema, as instantiated in working memory.



Figure 7: Pursuit Point and Envelope Zones

COSMODRIVE abilities for vehicle-control are thus supported in the Action module by the *pure-pursuit point method* (for monitoring the lateral and longitudinal position

of the car), and by the *envelope zones* strategies (for managing interactions with the other road users). Figure 6 illustrates this regulation strategy in the frame of a car-following task: the pursuit point determines the cap to be followed by the virtual ego-car, and the envelope zones are used for keeping a safe IVT distance with the lead car.

3.4 Simulation of visual distraction effects

By considering the empirical data presented in section 2, the visual scanning patterns of the additional screen collected during this experiment among human drivers (cf. fig 2) were implemented in the Perception module of COSMODRIVE, in order to simulate visual distraction effects on drivers' behaviors (visual strategies and vehicle control) and to investigate human errors liable to occur when drivers perform a visual secondary task while driving. Indeed, beyond the observable effects of visual distraction on drivers' performance, the aim of the COSMODRIVE computational modeling approach was also to simulate such distractive impacts on car drivers Situation Awareness.

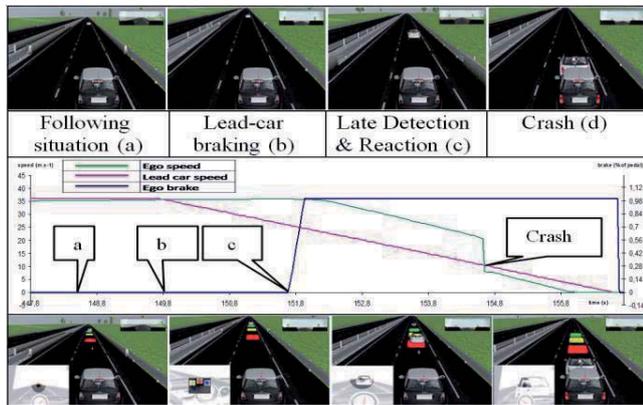


Fig. 8: driving performance simulation of a distracted driver

When driving, drivers must continually update their mental model of the driving situation as and when they dynamically progress on the road. In case of additional task requiring off-road scanning, mental model updating is imperfectly done and driver's actions are thus based on a mental representation that may dramatically differ of the situational reality, in case of a critical change in the traffic conditions.

This is typically what occurred in the example of crash initially presented in fig. 3, and then analyzed in Figure 8 and 9 from COSMODRIVE simulations. These 2 Figures correspond to a simulation case for a similar driving scenario presented in fig. 3 (free following task). Like 58 % of the observed human drivers, COSMODRIVE implemented here the first visual strategy for scanning of the additional screen (cf. fig. 2), requiring a long glance of 2 seconds. During these 2 seconds, the model manages the IVT with the lead car by using its mental representation of the driving situation (see *stages 2* on fig. 9). Unfortunately, the lead car brakes when the virtual eye is off-road and COSMODRIVE Situation Awareness progressively

becomes very different of the situational reality (stage 3 on Fig. 9). When the driver/model repays attention to the road scene (*view c* on Fig.8 and stage 4 on Fig. 9), they suddenly become aware of the critical gap between the expected lead car position (as mentally assessed during the off-road glance by assuming a steady speed of the lead car) and the critical nature of the objective reality (as illustrated at *stage 4* on fig. 9). Therefore, like the human driver presented in fig. 3, the model immediately carried out an emergency braking (reaction time of 0.8 sec. on Fig 8), but the crash cannot be avoided.

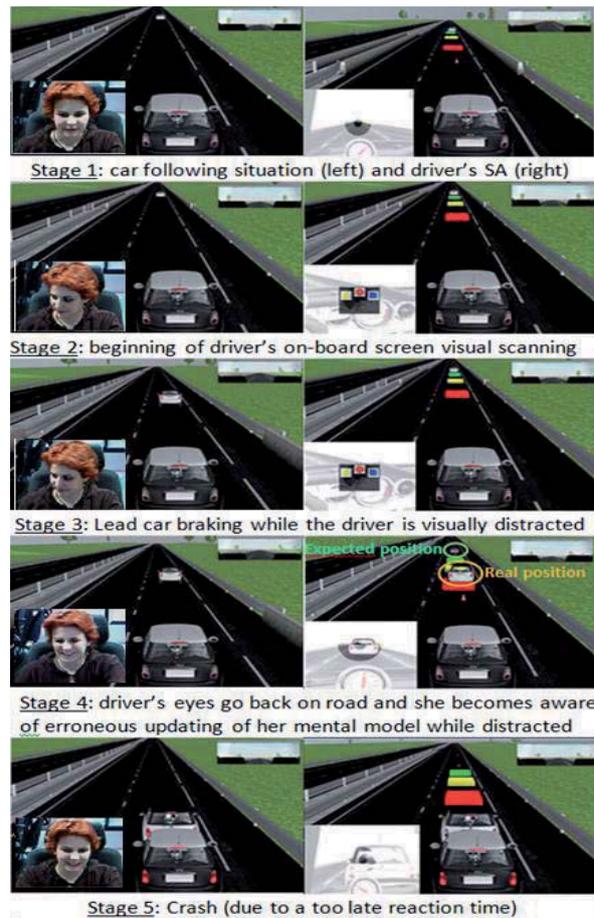


Figure 9: simulations of visual distraction effect on driver's SA

3.4 Conclusion and perspectives

As illustrated in Fig. 9, this type of simulation based on COSMODRIVE allow us to in-depth investigate and understand what happens in the driver's mind when visually distracted: incomplete or incorrect perception of roadway cues, due to off-road glances required by the secondary task, directly impacts the formulation of an adequate mental model (i.e. Situation Awareness) that will affect, in a second times, the decision making and the driving performance.

From these cognitive simulation abilities, it is expected in the future to explore visual distraction effects for a large set of driving scenarios, more particularly in terms of

inadequate mental model, that is of a crucial interest for analysing human errors at both behavioural and cognitive levels, or for explaining some involuntary risk taking of distracted drivers.

Acknowledgments

The research was supported by the European Commission 7th Framework Program, in the frame of ISi-PADAS Project.

References

- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, 111, (4), 1036-1060.
- Baddeley A.D. (1986). *Working Memory*. Clarendon Press.
- Bellet, T. (2011). Analysis, modeling and simulation of human operator's mental activities. In G. A. Boy (Ed), *Handbook of Human-Machine Interaction*, Ashgate.
- Bellet, T., Bailly-Asuni, B., Mayenobe, P., & Banet, A. (2009). A theoretical and methodological framework for studying and modelling drivers' mental representations. *Safety Science*, 47,1205–1221.
- Bellet, T., Bailly, B., Mayenobe, P., & Georgeon, O. (2007). Cognitive modelling and computational simulation of drivers mental activities. In P. Cacciabue (Ed.), *Modelling Driver Behaviour in Automotive Environment: Critical Issues in Driver Interactions with Intelligent Transport Systems*, 315-343, Springer Verlag.
- Bellet, T., Mayenobe, P., & Bornard, J.C. (2010). How to investigate the *living cognition*: an application to dynamic simulation of mental activities while driving. In D. D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the 10th International Conference on Cognitive Modeling* (pp. 1-6). Philadelphia, PA: Drexel University.
- Bornard J.C., Bellet T., Mayenobe P., Gruyer D., Claverie B. (2011). A perception module for car drivers visual strategies modeling and visual distraction effect simulation, *Proceedings of the 1st IEA-DHM Conference*, Lyon, June 2011.
- Donmez, B., Boyle, L. N., & Lee, J. D. (2007). Safety implications of providing real-time feedback to distracted drivers. *Accident; analysis and prevention*, 39(3), 581-90.
- Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. *Human Factors* 37, 32-64.
- Engstrom, J., Johansson, E., Ostlund, J. (2005). Effects of visual and cognitive load in real and simulated motorway driving. *Transportation Research Part F: Traffic Psychology and Behaviour* 8 (2), 97–120.
- Gruyer D, Royère C, du Lac N, Michel G, and Blosseville JM, 2006. Sivic and rt-maps interconnected platforms for the conception and the evaluation of driving assistance systems. In: *Proceedings of the ITS World Congress*. London, UK.
- Hall, E.T. (1966). *The hidden dimension*. New York: Doubleday.
- Kontaratos, N.A. (1974). A system analysis of the problem of road casualties in the United States. *Accident Analysis and Prevention*, 6, 223-241.
- Lee, J. D., Regan, M. A., & Young, K. L. (2009). What drives distraction? Distraction as a breakdown of multilevel control. In M. A. Regan, J. D. Lee, & K. L. Young (Eds.), *Driver distraction: Theory, effects, and mitigation* (pp. 42-56). Boca Raton, FL: CRC Press, Taylor & Francis Group.
- Mayenobe, P. (2004). *Perception de l'environnement pour une gestion contextualisée de la coopération Homme-Machine*. PhD Thesis, University of Clermont-Ferrand.
- Michon, J.A. (1985). A critical view of driver behavior models : what do we know, what should we do ?. In: Evans, L., Schwing, R.C. (Eds), *Human behavior and traffic safety*. New York: Plenum Press.
- Minsky, M. (1975). A Framework for Representing Knowledge. In P.H. Winston (Ed.), *The Psychology of Computer Vision*. New York: Mc Graw-Hill.
- Neisser U (1976). *Cognition and reality: principles and implications of cognitive psychology*. WH Freeman.
- Ochanine V.A. (1977). *Concept of operative image in engineering and general psychology*. In B.F. Lomov, V.F., Rubakhin & V.F. Venda (Eds), *Engineering Psychology*. Moscow: Science Publisher.
- Rasmussen J. (1986). *Information processing and human-machine interaction: an approach to cognitive engineering*. Amsterdam: North Holland.
- Senders, J.W., Kristofferson, A.B., Levison, W.H., Dietrich, D.W., Ward, J.L. (1967). The attentional demand of automotive driving. *Highway Research Record*, 195, 15–33.
- Schilder, P. (1950). *The image and appearance of the human body*. New York: International Universities Press.
- Schneider W., & Shiffrin R.M. (1977). Controlled and automatic human information processing I: Detection, search and attention. *Psychological Review*, 84, 1-88.
- Salvucci, D. D. (2001). Predicting the effects of in-car interface use on driver performance: an integrated model approach, *International Journal of Human-Computer Studies*, 55, 1, 85-107
- Salvucci, D. D. (2006). Modeling driver behavior in a cognitive architecture. *Human Factors*, 48, 362-380.
- Treat, J. R. (1980). A study of pre-crash factors involved in traffic accidents. *HSRI Research Review*, 10/11, 1-35
- Victor, T., Harbluk, J. L., & Engström, J. (2005). Sensitivity of eye-movement measures to in-vehicle task difficulty. *Transportation Research Part F: Traffic Psychology and Behaviour*, 8(2), 167-190.
- Wierwille, W.W. Huise M.C., Fischer T.J., Dingus T.A. (1988) Strategic use of visual resources by the driver while navigating with an in-car navigation display system, FISITA XXIIInd Congress. Automotive Systems Technology: the Future. Volume II. Technical papers Proceedings. Warrendale, PA : Society of Automotive Engineers, 2.661 – 2,675.
- Wierwille, W. W. (1993). Visual and manual demands of in-car controls and displays. In B. Peacock & W. Karwowski (Eds.), *Automotive Ergonomics* (pp. 299-320). London: Taylor & Francis.
- Zhang Y. (2011) *Visual and Cognitive Distraction Effects on Driver Behavior and an Approach to Distraction State Classification*. PhD Dissertation, North Carolina State University, Raleigh, USA.
- Zintchenko P. (1966). Quelques problèmes de psychologie de la mémoire. In A. Leontiev, A. Luria & A Smirnov (Eds), *Recherches Psychologiques en URSS*. Moscow: Editions du Progrès.

SimPilot: An exploration of modeling a highly interactive task with delayed feedback in a multitasking environment

Michael Schoelles (schoem@rpi.edu)

Cognitive Science Department, Rensselaer Polytechnic Institute
Troy, NY 12180 USA

Wayne D. Gray (grayw@rpi.edu)

Cognitive Science Department, Rensselaer Polytechnic Institute
Troy, NY 12180 USA

Abstract

Taxiing an airplane at a major airport requires the pilot to interact the world outside the cockpit, the instrumentations within the cockpit, and the co-pilot. As many actions require time to pass before their outcome can be evaluated, the pilot must have an approximate sense of how much delay should occur before the outcome of an action can be evaluated. Finally, taxiing is a paradigmatic example of multitasking. These three ingredients (a) a high level of interaction with dynamic task environments, (b) a sense of time, and (c) multitasking, present challenges for theories of cognition and the building of process models of taxiing. We describe a model, *SimPilot*, its initial validation, and its implications for cognitive theory.

Keywords: Multitasking; interactive behavior, delayed feedback, threaded cognition; cognitive control; task switching.

Introduction

A good applied problem drives basic science and a good theory can be useful, to paraphrase Newell (Newell & Card, 1985) and Lewin (1951). We believe such a relationship exists between taxiing an aircraft and cognitive theory. A task analysis of taxiing reveals that multiple cognitive, perceptual, and motor actions are preformed in parallel. The pilot must listen for commands and respond to commands from the ground controllers, while navigating the complex layout of airports. The pilot must steer and turn the plane while monitoring the aircraft speed. These and the other tasks require pilots to manipulate instruments in the cockpit while attending to signs and other planes outside the cockpit. In addition, key steps in taxiing require verbal and gestural contact with the co-pilot. The complexity of the Boeing 737-800 cockpit is evident in Figure 1.

In summary, taxiing is a paradigmatic example of multitasking where each task has its own subtasks that must be interleaved with those of the other tasks. It is exceedingly interactive. Not only does the pilot create change in the external and internal environment but many changes to which the pilot must respond arise from external factors. Finally, unlike most tasks modeled by cognitive science¹, changes produced by pilot actions may not be immediately apparent but requires the pilot to maintain some sense of

passing time and some expectation for when the results of his or her actions should become apparent.

SimPilot models the cognition and behavior of an airline pilot taxiing a Boeing 737-800. The mechanisms for multitasking that we use are Salvucci and Taatgen's theory of *threaded cognition* (2008; 2010), which at the moment is a candidate architectural mechanism within the ACT-R cognitive architecture (Anderson, 2007; Anderson, et al., 2004). We view *SimPilot* as an exploration of the strengths and weaknesses of the theory of threaded cognition as well as a potential tool for aviation psychology.

An open question for cognitive architectures is whether multitasking should be considered an architecture feature or a strategic adaptation that is driven by the accommodation of more basic architectural features to the demands of the task environment. Hence, besides being of applied interest, *SimPilot* may shed light on a key theoretical question.



Figure 1. The Boeing 737-800 cockpit.

Indeed, we suggest that *SimPilot* is the most comprehensive model yet developed using threaded cognition. Unlike prior models that are dual-threaded, *SimPilot* models three threads and uses the entire ACT-R 6.0 architecture to perform the taxiing application. In this paper, the next section will provide a brief overview of ACT-R 6.0 with threaded cognition. Then previous work within ACT-R to model multitasking performance is discussed. The *SimPilot* model is then described. We

¹ But not all tasks, see Anzai (1984), for an exception.

discuss the predictions the model made about pilot performance and how those predictions fared in an empirical evaluation. We conclude with an evaluation of threaded cognition as a mechanism to perform multitasking within a cognitive architecture. We end with a few words about the use of the simulated task environment, X-Plane™, for cognitive science research and computational cognitive modeling.

ACT-R 6.0

ACT-R 6.0 (Anderson, 2007) is an embodied cognitive architecture that has perceptual and motor components along with cognitive processing, memory, and control components. The perceptual and motor components enable SimPilot to operate user interfaces by passing the interface software the same commands passed by the input devices used by humans. As the SimPilot is a cognitive, not an artificial intelligence model, its input commands mimic the speed and accuracy of human users.

The ACT-R architecture requires the modeler to specify two types of knowledge. Declarative knowledge specified by the modeler represents background knowledge required to perform a certain task. The declarative knowledge is represented as chunks. A chunk has a type, which serves to specify the structure of the chunk. The chunk structure is composed of named slots, which hold values. Declarative chunks are stored in the Declarative Memory Module. Time to retrieve an item from memory varies as a function of the recency and frequency of that item's occurrence (Schooler & Anderson, 1997; Sims & Gray, 2004). Like humans, errors can occur in the memory retrieval process due to random fluctuations (noise) in memory strength or activation (Sims & Gray, 2004). Either the wrong chunk is retrieved or the intended chunk is not "strong" enough to be remembered.

The second type of knowledge specified by the modeler is procedural knowledge in the form of pattern matching productions. Productions specify how a certain task is done. A production consists of a set of constraints that must be satisfied before the actions specified by the production can be executed. Productions are stored in the Procedural Module. ACT-R checks every 50ms (human time) all of its productions and executes one of the productions whose pattern is matched. If more than one production can execute then ACT-R chooses the one it calculates would be the most useful at this time. This serial execution is not as constraining as it might seem and has been shown to be as accurate at simulating fine-grained human behavior as architectures that allow parallel firing of productions (Byrne & Anderson, 2001). If a production could fire but did not because another one had a higher utility, chances are that in 50 ms it will be able to fire. The productions are intended to represent the fine-grained procedural steps that are executed to perform some task. ACT-R adds noise to the utility calculation to simulate the variability in time and performance that humans make.

ACT-R maintains simulated human time in that time for ACT-R processes and actions are set to the theoretical times for the corresponding human events such as shift of visual attention, or memory retrieval. When the model does a task ACT-R produces a trace that includes the action taken and a time stamp. The trace allows model performance to be compared with human performance.

The perceptual components of ACT-R allow the model to see and hear. In common with the human brain, the visual component has *where* and *what* paths (Findlay & Gilchrist, 2003). The *where* path allows the model to detect features of an object such as color, size, and shape at a 2-D location in space. The *what* path moves visual attention to that location to encode the object with those features. ACT-R *hears* in much the same way that it sees in that sound events are detected and auditory attention is invoked to encode those sounds. By encoding objects and sounds in the environment the visual and auditory components add new declarative knowledge to the model. The motor component is the model's *hands* and *voice*. The manual component is capable of moving and clicking the mouse. Movement times are based on Fitts' Law (Fitts, 1954). The vocal module is capable of speaking text and subvocalization (see, e.g., Huss & Byrne, 2003).

The *imaginal* component of ACT-R is intended to hold intermediate representations required in solving a problem or performing some task. New declarative chunks can be added by this component. The *temporal* component maintains an internal clock. The *goal* component in hold chunks that guide task execution. For the model presented in this paper the default goal component is replaced with a module that implements a form of threaded cognition (Salvucci & Taatgen, 2008) that implements the multitasking required for the taxiing task.

Threaded Cognition

The current architectural candidate for multitasking in ACT-R is threaded cognition (Salvucci & Taatgen, 2008, 2011); an integrated theory of concurrent multitasking. Multitasking is defined as doing 2 or more tasks at once. A thread is sequence of processing steps coordinated by a serial procedural resource and executed across perceptual and motor resources. The key claims of threaded cognition are that multiple active goals can exist. Associated with each goal is a block of procedural processing. Processing conflicts can exist for procedural, declarative, perceptual, and motor resources. A thread will grab a resource if it needs the resource and the resource is available. It will release the resource when no longer needed. According to Salvucci and Taatgen, cognition favors the least recently processed thread. Declarative retrievals can be converted to hard coded productions over time thus reducing both declarative and procedural resource conflicts.

Background

Salvucci (Salvucci et al., 2006) explored interleaving task segments during a *discrete driving* task. Subjects used a

keyboard to steer a vehicle while entering navigation information as a second task. In particular they investigated how changing the timing characteristics of the driving task affected the interleaving of the tasks. To do this they used the temporal module of ACT-R and threaded cognition. They developed a top down and a bottom up model of cognitive control. In the bottom-up model, events (that is, changes to ACT-R's perceptual and temporal buffers) determine when task switching will occur. In the top-down model, the general executive (an early version of threaded cognition) interleaves the two tasks. Both models change the time interval for switching based on whether recent switches were early, late or on time.

Borst and Taatgen (2007) extended the threaded cognition concept that peripheral resources and declarative memory are shared between processes without executive intervention to problem representations, which are maintained in the imaginal buffer. They modified the discrete driving task of Salvucci (2006) to devise two tasks, one hard and one easy, in which the participants had to keep track of the problem state. The experiment and model they developed did show extra interference in task performance when both tasks needed the imaginal buffer.

Veksler (2011) used the latest implementation of threaded cognition in ACT-R 6.0 in a decision making task to monitor a task event while searching a display. This task is a spin-off of the Argus task (Schoelles, 2001) in which 20 targets appear on a radar screen. Each target has an assigned threat value, which the participant can acquire by clicking on the target. The task objective is to find the target with the largest threat value from a table of six alternatives, which is displayed on the right side of the screen. The degree of difficulty in acquiring the treat value is implemented via a lockout period, which is the time from clicking on the target to the actual display of the threat value. The lockout period was a between-subject condition that varied from 0 seconds to 8 seconds. Without any perceptual constraints her initial model switched many more times than human participants. When the perceptual constraint that the monitor task will only be initiated if model has not found a pre-attentive feature during the search thread was implemented, then the model did a much better at matching human switch rates.

Zemla (Zemla et al., 2011) has developed an ACT-R model of the taxiing task using the X-Plane simulation. The model does not use threaded cognition, since the focus of the model to produce a high-fidelity model of the turning and steering the plane while taxiing. The model is quite successful in modeling these subtasks. SimPilot does not steer or turn with the accuracy of this model but is more concerned with the interactive, multitasking aspects of the taxiing task.

SimPilot Description

The SimPilot model was developed as a proof-of concept system that intended to show that cognitive modeling can be applied to the evaluation of new technologies in aviation that are intended to increase runway safety. The system

consisted of the SimPilot model and ACT-R 6.0 running on one system, and the X-Plane simulation of a Boeing 737-800 running on the same or different computer. The communication between SimPilot and X-Plane simulator was via TCP/IP. Other aircraft running X-Plane on other computers simulated ground traffic at the airport. The X-Plane software provides several different interface options. One option is through a Software Development Kit (SDK). The user develops a plugin following the specifications of the SDK to access data variables used by X-Plane. Most of the data values that are displayed in the cockpit can be accessed both for reading and writing in this manner. The X-Plane system polls the plugin for requests to read and write these data values. For example, the ground speed of the aircraft can be read or the frequency of the radio can be read or set in this manner.

The scenario modeled by SimPilot begins with clearance from Air Traffic Control at the Dallas-Ft.Worth Airport, to taxi from the terminal, via a prescribed route to the *hold short* area of the runway. Once there, SimPilot must wait for clearance, then move onto the runway, and takeoff. Instructions and flight information can be given by Air Traffic control at anytime along the route. The route could involve several taxiways to reach the runway and other simulated aircraft. Concurrent subtasks include taxiing, monitoring the speed of the aircraft, maintaining situation awareness, and steering.

SimPilot Structure

SimPilot specifies visual-location chunks and visual-object chunks for the cockpit instruments. This file is read when the model is loaded. When the model moves visual attention to the location of an instrument the corresponding visual-object chunk is created.

ACT-R models are goal-driven, that is, task control is specified in declarative chunks that will serve as the current task goal. SimPilot specifies task goals for a number of subtasks, such as power-up, steering, turning, tune radio, switch lights, monitor speed etc. Since SimPilot uses multitasking, a task control chunk that specifies the subtasks that can execute together is also specified.

Productions have three structures. One structure does not have a goal but reacts to new information from the external environment. Examples include commands from ground control or comments from the co-Pilot. The structure of the task control productions specify what tasks can run concurrently. Third, regular productions have a goal with either a state slot or a time constraint slot.

SimPilot Task Control Flow

The first version of SimPilot did not use threaded cognition, but attempted to implement multitasking by using ACT-R's default ACT-R, one-level deep, goal buffer in several different ways. It was found that this switching was either not very cognitively plausible or took too much time. In one approach, the old goal was stored in a slot in the new goal and retrieved directly from that slot when it was necessary

to switch back. This approach is not cognitively plausible or very flexible with more than two goals. Another approach is to retrieve old goals through declarative memory (Altmann & Gray, 2008) but this approach did not meet the time requirements of a highly interactive environment requiring immediate actions. Threaded cognition offered a good alternative although all models using threaded cognition up to this point have been far simpler than the complexity required for taxiing. Task analysis also showed that the number of tasks that are being done at one time varies. The threaded cognition goal module maintains a set of goals. The goal buffer is used as part of threaded cognition and in accordance with Taatgen's Minimal Control Principle (Taatgen, 2007), using a goal state slot is kept to a minimum. The control constraints come from the availability of the perceptual modules. In addition, the temporal module and buffer are used to monitor events at specific intervals, which also provides a form of control.

We define a subtask (of taxiing) to be a set of one or more goals to be executed using threaded cognition. That is, the goal module is responsible for switching the goal buffer between these goal chunks. The subtask control chunks are linked together through a slot which contains the next subtask to be executed. When the current goal set has accomplished its part of the task a subtask control chunk is made the current goal, which retrieves the chunk specifying the next multitasking set.

In SimPilot, a thread maintains control by either not clearing the perceptual buffers or having the imaginal buffer maintain a representation unique to the thread. For the most part a thread can only start executing if the perceptual and imaginal buffers are clear. A thread gives up control by clearing these. As the model was being developed it was realized that communication between threads is sometimes required. Normally in ACT-R communication between productions is done through the imaginal buffer, but in SimPilot the imaginal buffer is thread specific so this pointed to the need for another buffer, which can be considered as an extended imaginal buffer that is common to all threads. In SimPilot this is called the *situation buffer*.

SimPilot Steering When humans steered X-Plane, they used a joystick that had a nose wheel control capability. For the model to steer X-Plane several options are available. Early on in the project the joystick was configured to change the yoke pitch, roll, and yaw, so initially the option that sets the SDK variables for the yoke pitch, roll, and yaw was implemented. When the joystick was reconfigured to manipulate the nose wheel, the option that the mouse acts as the joystick and changes the nose wheel in the same way as the human subjects was implemented. Also, steering requires both perception and manual operations. SimPilot looks at the heading display on the cockpit to monitor the current heading and looks at a point on the windshield, which is the target for the movement.

It uses the temporal module to keep from responding too fast to course corrections.

SimPilot Turning Turning is similar to steering, but in turning the plane goes from a beginning heading to a final heading, and the current heading changes rapidly. Also, turns decrease the momentum of the aircraft so human pilots often increase thrust during the second half of the turn. In the first half of the turn SimPilot steadily increases the deflection of the nose wheel in the direction of the turn, it then brings it back to zero deflection in the second half, always trying to keep the rate of turn below 5 degrees/second. SimPilot looks at a point on the windshield and moves the cursor to that location.

SimPilot Parameters The model is predictive in the sense that it was not fit to data. The standard ACT-R parameters were used. It would be very hard to fit the model to data for individual parameters due to the complexity of the task. Likewise, as the model requires the X-Plane simulation and can only be run in real time, each model run takes 5-10 minutes. This real-time constraint makes it nearly impossible to do the 100's or 1000's of model runs that most model fitting requires.

SimPilot Representation of Environment

Interactive models like SimPilot are heavily dependent on a good representation of the environment. The cockpit that X-Plane provides (as shown in Figure 1) is very complex and contains many user interface objects that are not the usual Human Computer Interaction (HCI) type of objects. For example, tuning the radio, which is one of the capabilities of the model, requires an interactive routine of 22 productions. The radio has an inner and outer dial. The outer dial determines the integer portion of the frequency and the inner dial controls the decimal portion. The radio has an active indicator and a standby indicator. The model sets the standby indicator and then presses a button to switch the frequency entered to the active display.

To navigate the airport or follow a route, the locations of airport signs is required. Again X-Plane does not provide an automatic way to encode this information. We used Google Earth to obtain some of the taxiway and runway locations. KML files were exported from Google Earth and the MSS contains some code to parse these files.

SimPilot Performance Measures

Table 1 shows the number of visual attention shifts and the number of subtask switching for the speed monitoring, navigation monitoring and steering threads for two taxiing tasks. These results were extracted from the ACT-R trace files produced for each run. Due to lack of eye data we cannot compare these to human saccades and fixations.

Table 1. Model Attention and Subtask Shifts.

Measure	Taxi to 35L (564 secs)	Taxi to 35R (738 secs)
Visual Attention Shift	1042 (1.8/s)	1362 (1.8/s)
Speed Monitoring	381 (0.7/s)	409 (0.6/s)
Navigation Monitoring	432 (0.8/s)	568 (0.8/s)
Steering	432 (0.8/s)	566 (0.8/s)

SimPilot Validation

Human data was collected from 6 pilots with varying degrees of experience. The model has only gone through an alpha where no model parameters were fit to the data, so the model data presented in Table 2 (Jungemann, 2011) can be interpreted as model predictions. In fact the model was only run in the complete system configuration once, which did expose several problems.

Table 2. Performance measurements and results

Performance Measure	Human Mean/SD	Model Mean/SD
Average Taxi Speed	9.8/1.4	6.5/0.7
Maximum Taxi Speed	15.3/2.6	8.7/0.5
Average Throttle Setting	0.13/0.03	0.11/0.002
Maximum Throttle Setting	0.46/0.21	0.15/0.01

The data shows that this version of the model is far from being an expert pilot, but several factors make quantitative comparison of model and human data difficult. The braking capability of the human pilots was very different than the model. The humans had pedals that were used to brake the plane, but ACT-R has no foot motor module or interface to the pedals so the model performed braking through adjusting a X-Plane variable. The human pilots used a joystick to steer the aircraft via the nose wheel of the plane. The model could not control the joystick directly in X-Plane but used the cursor to control the nose wheel.

Some of the measures that we could compare model versus human performance are shown in Table 2. The average taxi speed is the aircraft ground speed in nautical miles per hour, calculated from start of taxi to the hold short area of the runway. The maximum taxi speed is the highest ground speed attained on taxiway. The average throttle setting is X-Plane data value that ranges from Idle to Full. The maximum throttle setting is the highest throttle setting from start to hold short.

Discussion

Since we do not have actual data on the task switching behavior of human pilots, novice or expert, taxing with X-Plane, our evaluation of threaded cognition is from an engineering perspective. By itself the implementation of threaded cognition is underspecified for multitasking in this environment. The main issue for cognitive engineering is how much does threaded cognition shift the burden of task switching from the modeler to the architecture. In the ACT-R architecture without threaded cognition the modeler must make an explicit request of the goal module to switch goals. With threaded cognition the goal module, if two or more goals match, will allow only one production with a goal of that type to execute at a time, since it will be placed at the end of the queue after its production executes. This will be fine in situations where this type of alternate behavior is required, but in most situations other factors determine task behavior. For example, pilots do not alternate between checking the speed of the plane and looking at the center line. The pilots check their position on the taxiway much more than checking the speed. The speed is checked at periodic intervals, which is implemented in SimPilot by the temporal module.

In many cases, the first production of a thread is a check for some condition, for example, is the speed of the aircraft within certain limits. If not, then the function of the thread is to correct that condition as fast as possible. In these cases the thread should not be interrupted. With the current implementation, it is up to the modeler to code this into the productions. So while some of the task control has been shifted to the architecture, much of it must still be done by the modeler.

In developing the model, it became evident that decomposing the taxiing task into subtasks required that each subtask maintain its own representation in the imaginal buffer to hold the knowledge unique to that subtask. For example, monitoring speed requires knowledge about speed limits while navigating requires knowledge about position. Global data, such as where you are on the route to the runway, is data shared by different subtasks and need to be held in a shared buffer. In SimPilot a global buffer called the situation buffer was created. This buffer can be thought of as an imaginal buffer for the entire task. It has the same modeling considerations as regular imaginal buffers such as how long should the data persist before some it should be practiced, etc.

The interface to the X-Plane environment initially seemed promising but has not worked as hoped. For the cockpit interface, SimPilot needs to know the x and y pixel location, name, color and size of all the instruments. X-Plane does provide the names and the x and y pixel locations for some of the instruments but not for the majority of them. This information has to be hand calculated which is a very labor-intensive process. The reactivity of the model to changes in the plane's data values is constrained by the polling interface. X-Plane does provide a datagram interface, which would allow faster interactions between the model and X-

plane, but this interface is not well-documented nor guaranteed not to change.

In order to be able to make comparisons between SimPilot and human pilots, an analysis of where humans are looking while they are doing this task is essential. The next step in the development of SimPilot is to collect eye data on human multitasking behavior in this task and use the results improve the model.

To simulate the multitasking required to taxi a Boeing 737-800 at a major airport, the SimPilot model uses all the components of ACT-R with an implementation of the theory of threaded cognition. This effort has both theoretical implications for ACT-R and threaded cognition and is a start on solving the important applied problem of the effects of multitasking on pilot performance.

Acknowledgments

This work was supported by NASA Ames Research Center through a Small Business Innovative Research Contract (Contract No. NNX09CB45C) awarded to Aptima, Inc. All opinions expressed in this article are those of the authors and do not necessarily reflect the official opinion or position of Rensselaer Polytechnic Institute, Aptima, Inc., or NASA. We would like to thank Dr. James Johnston and Dr. Michael Feary at NASA Ames Research Center for funding this research. We also would like to thank Gabe Ganberg, Jeff Jungemann, Joe Thoreson and Jamie Estock of Aptima, Inc.

References

- Altmann, E. M., & Gray W. D. (2008). An integrated model of cognitive control in task switching.. *Psychological review*, 115(3), 602-39.
- Anderson, J. R. (2007) How Can the Human Mind Occur in the Physical Universe? New York: Oxford University Press.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. L. (2004). An integrated theory of the mind. *Psychological Review*, 111(4), 1036-1060.
- Anzai, Y. (1984). Cognitive control of real-time event-driven systems. *Cognitive Science*, 8, 221-254.
- Byrne, M. D., & Anderson, J. R. (2001). Serial modules in parallel: The psychological refractory period and perfect time-sharing. *Psychological Review*, 108(4), 847-869.
- Byrne, M. D., & Kirlik, A. (2005). Using computational cognitive modeling to diagnose possible sources of aviation error. *International Journal of Aviation Psychology*, 15, 135-155.
- Byrne, M. D., Kirlik, A., & Fleetwood, M. D. (2008). An ACT-R approach to closing the loop on computational cognitive modeling: Describing the dynamics of interactive decision making and attention allocation. In D. C. Foyle & B. L. Hooey (Eds.), *Human performance modeling in aviation* (pp. 77-104). Boca Raton, FL: CRC Press.
- Findlay, J. M., & Gilchrist, I. D. (2003). *Active vision: The psychology of looking and series*. New York: Oxford University Press.
- Fitts, P. M. (1954). The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement. *Journal of Experimental Psychology*, 47(6), 381-391.
- Huss, D. G., & Byrne, M. D. (2003). An ACT-R/PM model of the articulatory loop. In F. Detje, D. Dörner & H. Schaub (Eds.), *Fifth International Conference on Cognitive Modeling* (pp. 135-140). Bamberg, GE: Universitas-Verlag Bamberg.
- Jungemann, J. (2011).Aptima Final Report: Computational Model and Measurement Tool for Evaluating the Design of Flight Deck Technologies II.
- Lewin, K. (1951). *Field theory in social science*. New York: Harper Row.
- Newell, A. & Card, S. K. (1985). The prospects for psychological science in human-computer interaction. *Human-Computer Interaction* 1, no 3, 209-242.
- Salvucci, D. D., & Taatgen, N. A. (2008). Threaded cognition: An integrated theory of concurrent multitasking. *Psychological Review*, 115(1), 101-130.
- Salvucci, D. D., & Taatgen, N. A. (2011). *The multitasking mind*. New York: Oxford University Press.
- Schoelles, M. J., & Gray, W. D. (2001a). Argus: A suite of tools for research in complex cognition. *Behavior Research Methods, Instruments, & Computers*, 33(2), 130-140.
- Schooler, L. J., & Anderson, J. R. (1997). The role of process in the rational analysis of memory. *Cognitive Psychology*, 32(3), 219-250.
- Sims, C. R., & Gray, W. D. (2004). Episodic versus semantic memory: An exploration of models of memory decay in the serial attention paradigm. In M. C. Lovett, C. D. Schunn, C. Lebiere & P. Munro (Eds.), *6th International Conference on Cognitive Modeling* (pp. 279-284). Pittsburgh, PA: Carnegie Mellon University/University of Pittsburgh.
- Taatgen, N. A. (2007). The minimal control principle. In W. D. Gray (Ed.), *Integrated models of cognitive systems*. New York: Oxford University Press.
- Veksler, B. (2011) Modeling the Visual Search Process. Unpublished Doctoral Thesis, Rensselaer Polytechnic Institute
- Zemla, J.C., Ustun, V., Byrne, M.D., Kirlik, A., Riddle, K., & Alexander, A.L. (2011).An ACT-R Model of Commercial Jetliner Taxiing.Proceedings of the 55th Human Factors and Ergonomics Society.

Cognitive Re-Use via Emergic Networks

David Pierre Leibovitz (dpleibovitz@ieee.org) &
Robert West (robert_west@carleton.ca)

Institute of Cognitive Science, 2201 DT, Carleton University
1125 Colonel By Drive, Ottawa, Ontario, K1S 5B6 Canada

In this paper we introduce a new cognitive modeling system called Emergic Networks. The Emergic Network system is designed to facilitate functional, nonlinear decomposition with the aim of understanding how different neural systems can interact to produce specific instances of cognitive functionality. The first part of the paper briefly describes the motivation for the system and the second part briefly describes the system and provides a web location for downloading.

Second Order Emergence

The type of emergence we are interested in involves *cognitive re-use* (Anderson, 2010), where the same neural circuits interact in different ways to produce different types of cognitive functionality. That is, the parts, taken together, cannot be considered as a module devoted to a specific cognitive function. There are two ways that emergence can be thought of as underlying cognitive functions. The first is when a cognitive function arises as an emergent property from a dedicated system of neural circuitry. In this case there is a one to one mapping from cognitive function to neural structure. We can call this, *neural to functional emergence*. Attempting to localize brain areas associated with particular cognitive functions would be an example of research based on this assumption (Poldrack, 2010). However, it is also possible for functions directly supported by dedicated neural circuitry to interact and produce *second order functional emergence*; that is, an emergic function arising from the interaction of underlying functions. In the case of second order functional emergence, it would be a mistake to search for a dedicated neural structure designed to produce that function. Instead, the goal would be to explain how such functions emerge through the interaction of underlying functions.

Everyone agrees that some form of neural to functional emergence allows the brain to act in a functional way. However, second order functional emergence is controversial because it implies that the industry of mapping high-level observable functions to specific brain areas is partially or wholly misguided. This issue is represented by two opposing theoretical positions: the *Anatomical Modularity* position and the *Cognitive Re-Use* position. According to the Anatomical Modularity position, each cognitive function is implemented by a dedicated neural system (Bergeron, 2007). In contrast, the Cognitive Re-Use position (Anderson, 2010) asserts that most or all cognitive functions are the product of underlying, interacting functions that can play different roles in the formation of

different cognitive functions. Theories related to the idea of Cognitive Re-Use include: *neural exploitation, shared circuits model, neuronal recycling, massive redeployment, highly connected hubs, descent-with-modification modularity, the Lego model, fine-grained information processing operations* and *distributed processing* as mentioned by Anderson.

In terms of emergence, the anatomical modularity position seems to be associated with an implicit assumption that the modules will interact in an additive, linear manner to produce easily decomposable aggregate behaviours. In contrast, the cognitive re-use position seems like it would almost require some form of non-linear interaction or emergence to get it to work. This difference is possibly due to the fact that the assumed output of anatomical modules is often symbolic in nature, whereas the functions in cognitive re-use would be primarily pre-symbolic; that is, functions that need to be combined to produce the ability to process symbols (or to act as if we can process symbols).

Some cognitive modeling systems include recurrent interactions that can lead to emergence. For example, recurrent networks employ recurrent feedback loops, CLARION (Helie & Sun, 2011) has explicit modules for top down and bottom up processes that can potentially produce recurrent feedback, and ACT-R has been used to model recurrent feedback between agents (e.g., West, Stewart, Lebiere, & Chandrasekharan, 2005). Recurrent feedback also plays an important role in Dynamic systems models of cognition (Krech, 1950) and in various neural models, such as NENGO (Eliasmith & Anderson, 2003). However, while all of these systems can employ feedback to achieve interesting emergent effects, none of these systems were designed specifically to model and study the role of second order emergence in cognition. Dynamic systems theory is designed for studying emergence, but it is not a modeling system. It is a collection of mathematical tools for analyzing dynamic systems. A dynamic systems model must be constructed mathematically and there are no cognitive or neural constraints on how this should be done. Spiking neural models are designed to model neural to cognitive emergence. Such systems can be used to model second order functional emergence but the process would be guided and constrained by bottom up, neural constraints. This is a good thing, but a more complete research program would also involve exploring this from a purely functional point of view, as we are not yet completely sure what the neural constraints should be.

The emergic network is designed to explore how lower level functions are combined and re-used to produce

multiple instances of higher level functionality. Emergic networks are non-symbolic and in some ways similar to spiking neuron models. In particular, the emergic units that make up the networks are similar to clusters of neurons that perform a specific function and the connections between the emergic units are functionally similar to the neural connections between clusters. Also, similar to neural systems, emergic networks process information in a continuous manner. However, the point of emergic networks is to understand emergence based on functionality, not neural behaviour. As far as we know, the emergic network is the first cognitive modeling system specifically designed to model second order functional emergence and re-use.

Emergic Networks

The decomposition of intelligent behaviour into cognitive functions and structures has traditionally progressed by hypothesizing macro-level functions that emerge from a minimal interaction between localized brain modules. An alternative approach considers finer grained brain modules as realizing micro-level functions that are extensively reused (M. L. Anderson, 2010) and interact to cause higher order functions to emerge, often in a non-linear fashion. However, this latter approach greatly complicates the scientific reduction of cognition not only because of an extra level of non-linear decomposition, but mostly due to a lack of characterization and experience in such an analytical space. Emergic networks are meant as a step forward in clarifying and dealing with this issue.

An *emergic network* consists of a connected set of *emergic units*, each forming a micro-level portion of functional computation and behaviour. An emergic unit computes a function that can be represented mathematically or by computer code. Emergic units have input and output *ports* that connect them to other emergic units through *links*. Links transport *values* between the units. The values can take any mathematical form (e.g., numbers, vectors, and statistics). Links are unidirectional and have a scaling factor that is set to 1 by default. Input and output ports can be connected to multiple links. By default, link values are summed at input ports while an output port will duplicate its value to all destinations.

The emergic network architecture is synchronous, with links having a minimal delay of one tick. That is, the delivery of values through the links is clocked so that all values arrive at the same time. The values flowing around the network are intended to represent small changes, i.e., to approximate a physical system of continuous change and interaction (Rumelhart, McClelland, & Group, 1987). Emergic networks model asynchronous behaviour by setting time delays (counted in ticks) small enough for computing the effective functions of emergic units in an incremental fashion.

It is interesting to note that although the emergic network was developed independently, the structure we have just described is very similar to NENGO, which is a spiking neuron modeling system. This is can be attributed to the fact

that both systems are concerned with identifying the basic units of neural computation. In NENGO the functions carried out by the emergic units are carried out by realistic spiking neuron models. However, the focus of these systems is different. The focus of NENGO is to consider realistic neural constraints when modeling systems of neural computation. The focus of emergic networks is to model second order emergence and cognitive re-use. That is, to explore how different functions can interact to produce cognitive phenomena. The goal with emergic networks is to work out cognitive design principles that might otherwise be overlooked by assuming a one to one mapping between cognitive functions and neural units.

The code for building Emergic Networks can be downloaded from http://emergic.upwize.com/?page_id=6. Currently we are working on an emergic network model to produce a unified account of low level visual effects such as filling-in. Preliminary results are available at http://emergic.upwize.com/?page_id=31.

References

- Anderson, M. L. (2010). Neural reuse: A fundamental organizational principle of the brain. *Behavioral and brain sciences*, 33(4), 245-66.
- Bergeron, V. (2007). Anatomical and Functional Modularity in Cognitive Science: Shifting the Focus. *Philosophical Psychology*, 20(2), 175-195.
- Eliasmith, C., & Anderson, C. H. (2003). *Neural Engineering: Computation, Representation, and Dynamics in Neurobiological Systems* (p. 376). Cambridge, MA: MIT Press.
- Helie, S., & Sun, R. (2011). How the Core Theory of CLARION Captures Human Decision- Making. *Proceedings of the International Joint Conference on Neural Network* (pp. 173-180). IEEE.
- Krech, D. (1950). Dynamic systems, psychological fields, and hypothetical constructs. *Psychological Review*, 57(5), 283-290.
- Poldrack, R. A. (2010). Mapping Mental Function to Brain Structure: How Can Cognitive Neuroimaging Succeed? *Perspectives on Psychological Science*, 5(6), 753-761.
- Rumelhart, D. E., McClelland, J. L., & Group, P. R. (1987). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations* (p. 567). Cambridge, MA: MIT Press.
- West, R. L., Stewart, T. C., Lebiere, C., & Chandrasekharan, S. (2005). Stochastic Resonance in Human Cognition: ACT-R Versus Game Theory, Associative Neural Networks, Recursive Neural Networks, Q-Learning, and Humans. In M. Bara, B., Barsalou, L., & Bucciarelli (Ed.), *Proceedings of the 27th Annual Meeting of the Cognitive Science Society* (pp. 2353-2358). Austin, TX: Lawrence Erlbaum.

Hierarchical Bayesian Modeling of Manipulation Sequences from Bimodal Input

Alexandra Barchunova (abarch@cor-lab.uni-bielefeld.de)

Bielefeld University, Cor-Lab, Universitaetsstrasse 21-23
Bielefeld Germany

Jan Moringen, Robert Haschke, Helge Ritter

Bielefeld University, Universitaetsstrasse 21-23
Bielefeld Germany

Abstract

We propose a hierarchical approach for Bayesian modeling and segmentation of continuous sequences of bimanual object manipulations. Based on bimodal (audio and tactile) low-level time series, the presented approach identifies semantically differing subsequences. It consists of two hierarchically executed stages, each of which employs a Bayesian method for unsupervised change point detection (Fearnhead, 2005). In the first step we propose to use a mixture of model pairs for bimanual tactile data. To this end, we select "object interaction" and "no object interaction" regions for the left and the right hand synchronously. In the second step we apply a set of Autoregressive (AR) models to the audio data. This allows us to select regions within "object interaction" segments according to qualitative changes in the audio signal. Two simple model types that allow the calculation of modality-specific segment likelihoods serve as a foundation for this modeling approach. Based on the acquired ground truth, empirical evaluation has showed that the generated segments correctly capture the semantic structure of the test time series. The developed procedure can serve as a building block for higher-level action and activity modeling frameworks.

Introduction

An important objective of today's interdisciplinary research of human-machine interaction is machine perception of human action and activity (Krüger, Kragic, Ude, & Geib, 2007), (Bobick & Davis, 2001), (Turaga, Chellappa, Subrahmanian, & Udrea, 2008), (Aggarwal & Park, 2004). Areas like cognitive and social robotics, artificial intelligence, ambient intelligence, sports science and neurobiology collaborate on understanding of the mechanisms of human movements. Research in cognitive robotics is aimed towards enabling robots to interact with humans in everyday scenarios. Within this area, we focus on the topic of autonomous identification of bimanual object manipulations from low-level bimodal observation sequences. In order to participate in a simple interaction scenario or learn from a human, a robot needs the ability to autonomously single out relevant parts of the movement executed by a human.

Analysis of various sensor readings describing the human hand dynamics during manual interaction have been conducted recently by different researchers (Bernardin, Ogawara, Ikeuchi, & Dillmann, 2005; Dillmann, Rogalla, Ehrenmann, Zöllner, & Bordegoni, 2000; Kawasaki, Nakayama, & Parker, 2000). In general, one is interested in autonomous identification of action primitives in the context of imitation learning and human-machine interaction (Sanmohan, Krüger, & Kragic, 2010; Takano & Nakamura, 2006). Within this domain, Matsuo et al. focused on force feedback (Matsuo,

Murakami, Hasegawa, Tahara, & Ryo, 2009) while a combination of different sensors like CyberGlove, Vicon or magnetic markers and tactile sensors has been used by (Pardowitz, Knoop, Dillmann, & Zöllner, 2007), (Kawasaki et al., 2000) and (Li, Kulkarni, & Prabhakaran, 2006). In (Zöllner, Asfour, & Dillmann, 2004) a bimanual approach is described. Audio and ultra-wide band tags have been successfully used in (Ogris, Stiefmeier, Lukowicz, & Troster, 2008) and (Ward, Lukowicz, Troster, & Starner, 2006).

Identification and learning of manual action primitives from continuous sequences is still an open question. We address it by proposing a novel hierarchical approach for uni- and bimanual time series. The bimodal approach is inspired by the fact, that humans employ different perception channels like hearing, proprioception, haptics and vision. Furthermore, our recent work has showed that audio and tactile data can generate a symbolic sequence of action primitives of sufficient granularity and semantic content (Barchunova, Haschke, Franzius, & Ritter, 2011). The automatically extracted action primitives have been successfully used in a classification application with HMM-based models (Grossekathöfer et al., 2011). During a considerable number of simple object manipulations (e.g. grasping, shifting, shaking, pouring, stirring or rolling) application of force is naturally accompanied by specific types of sound. We exploit this fact by performing modeling and segmentations based on the analysis of the audio signal structure and contact forces recorded on the fingertips. Our method handles three main challenges arising in automated modeling of action sequences: (i) inter- and intrapersonal variance of sensor data, (ii) absence of prior knowledge about the structure of the action sequence (i.e. location, type and number of action primitives) (iii) modality fusion. The data recorded for different human demonstrators exhibits a high degree of interpersonal and intrapersonal variance. However, our method solely depends on the temporal structure of the data and is invariant to absolute data values, the speed of action execution, way of grasping or the manipulation object. Furthermore, the output is person-invariant. Our method does not employ any specific knowledge about the components of the action sequence. Based on two simple models, the modeling does not require a large set of domain-specific heuristics describing each action primitive as is commonly the case in similar approaches (Pardowitz et al., 2007; Kawasaki et al., 2000; Zöllner & Dillmann, 2004). Due to the simplicity of these two fundamental



Figure 1: Experimental setup: a human demonstrator wearing contact sensors performs manipulation operations with a plastic bottle equipped with a structure-borne microphone. The CyberGlove trajectories also recorded in this scenario are not evaluated in this work.

models and the modeling concepts used within our approach, the developed procedure can be easily used in a wide range of scenarios, like imitation learning, cooperation and assistance. Because the segmentation steps for individual modalities are executed hierarchically, no additional multimodal fusion (e.g. (Ogris et al., 2008)) is necessary.

We evaluate our method in an everyday scenario in which a human demonstrator performs several object manipulation operations with a large non-rigid plastic bottle with a handle. In this evaluation, we assess the performance of the segmentation method w.r.t. the accuracy of the generated segment borders. The rest of this paper is organized as follows: Sec. "Experimental Setup" explains the acquisition of action sequences within the scenario. Sec. "Segmentation Method" introduces the two steps of the proposed method. In Sec. "Evaluation" we discuss our evaluation method and experimental results of the procedure, Sec. "Conclusion and Outlook" concludes the paper with a brief discussion and outlook.

Experimental Setup

In our scenario, a human demonstrator performs sequences of simple uni- and bimanual object manipulations with a gravel-filled plastic bottle¹, as can be seen in Fig. 1.

We use two types of sensors to record the time series of the performed action sequences (corresponding modality names used in formulas appear in parentheses):

- A structure-borne microphone AKG C411 L attached to the bottle records an audio signal (a), which is focused on in-object generated sound, ignoring most environmental noise.

¹The use of gravel instead of liquid is due to safety concerns. We have used liquids in a similar scenario restricted to the audio modality.

- 2×5 FSR pressure sensors attached to the fingertips of each CyberGlove (t : both hands, t_l : left hand, t_r : right hand) record the contact forces.

The human demonstrator was instructed to perform a sequence of basic manipulation actions in the fixed order showed in the enumeration below. To obtain ground truth for later evaluation of computed segment borders we have used two methods of ground truth acquisition: manually annotated (*unconstrained*) and automated cue-driven (*constrained*). In the *unconstrained* scenario the human demonstrator was asked to conduct the sequence at her/his natural speed. The annotation of the action sequences has been conducted based on a video recording of the interaction scene. Within the cue-driven *constrained* scenario the aspired beginning or end of an action within a sequence was signalled to the human demonstrator via headphones as explained in (Barchunova, Haschke, Franzius, & Ritter, 2011). To achieve a rich variance for individual action primitives between different trials in the *constrained* scenario, we added Gaussian noise to the nominal time of the action primitives as specified in parentheses:

1. pick up and hold the bottle with both hands ($2 s + \eta_1$)
2. shake the bottle with both hands ($.7 s + \eta_2$)
3. hold the bottle with both hands ($.3 s + \eta_3$)
4. put down the bottle and pause ($1 s + \eta_4$)
5. unscrew the cap with both hands ($1.2 s + \eta_5$)
6. release cap and pause ($1 s + \eta_6$)
7. grasp and lift the bottle with right hand ($2 s + \eta_7$)
8. pour with right hand ($1 s + \eta_8 + 1 s + \eta_9$)
9. hold the bottle ($.3 s + \eta_{10}$)
10. put down the bottle and pause ($1 s + \eta_{11}$)
11. screw the cap with both hands ($1.2 s + \eta_{12}$)

The random variables $\eta_i \sim \mathcal{N}(0, .5 s)$ denote the randomized timing of subsequences. The overall length of the time series of a trial accumulates to approximately 30 seconds. Both annotation methods have specific advantages and disadvantages. The cue-driven annotation is a completely automated way of ground truth acquisition, avoiding time-consuming manual annotation but putting constraints on the execution speed and sequence. This method is suitable for acquiring ground truth for large number of trials. Manual annotation is more precise and more time-consuming, but it does not rely on perfect adherence to audio cues by the human demonstrator.

Segmentation Method

The recorded time series of multiple sensors capture complex descriptions of action sequences. This section describes how such time series data is segmented and modeled. Our segmentation approach applies Fearnhead’s method (Fearnhead, 2005) previously used for unsupervised detection of multiple change points in one-dimensional time series. In his work Fearnhead describes a deterministic method that maximizes the posterior distribution of the number and location of change points w.r.t given observations. The estimated change points are optimal in the sense that a combination of a prior distribution on segmentations and segment-wise likelihoods is maximized. The segment likelihood is computed with respect to a single model chosen from a fixed set of models. Our approach combines a preprocessing step with a set of simple, modality-specific models to enable a probabilistic description of the recorded time series as the basis for applying Fearnhead’s algorithm (introduced in Sec. “Bayesian Segmentation”). In the preprocessing step each sensory channel is reduced to a compact scalar description capturing its temporal structure. Two basic data models, an autoregressive and a threshold model, are employed within the procedure for the channel-specific modeling (see Sec. “Basic Data Models”). Their association to the tactile and audio modalities is explained in Sec. “Signal Preprocessing for Basic Models”. Finally, in the Sec. “Two-Stage Segmentation” the two stages of the procedure – the segmentation based on the tactile modality and the subsegmentation based on audio – is described.

Bayesian Segmentation

In general, Fearnheads’ algorithm segments an arbitrary time series $y_{1:T}$ by determining a set of change points $1 < \tau_1 < \dots < \tau_N < T$ at which qualitative changes occur in the data. Within the probabilistic framework of Fearnhead’s algorithm, the optimal segmentation is obtained by maximizing the Bayesian posterior² $P(y_{1:T} | \tau_{1:N})P(\tau_{1:N})$ which consists of a likelihood term and a prior distribution over segmentations $P(\tau_{1:N})$. In a common choice of this prior, the probability $P(\tau_{1:N})$ is composed of probabilities of individual segment lengths which are computed according to the geometric distribution $P(l) = \lambda(1 - \lambda)^{l-1}$. Consequently, the prior is characterized by a single parameter λ that is reciprocal to the expected segment length under a geometric distribution, i.e. $\lambda \propto 1/u$ where u is the expected length of subsequences. Once λ has been chosen, neither the number of change points N nor any information regarding their positions have to be specified in advance.

The likelihood term $P(y_{1:T} | \tau_{1:N})$ is the probability, that the observed time series originates from a set of given models, which are fixed over the period of an individual segment. To this end, a finite set of models \mathcal{M} is employed. Given a particular model $m \in \mathcal{M}$, the marginal likelihood $P(y_{s:t} | m)$ is the probability, that the entire subsequence $y_{s:t}$ can be ex-

²We suppress the constant normalization factor $P(y_{1:T})^{-1}$.

plained by this model. Prior probabilities $P(m)$ can be associated with all models to reflect their relative frequency.

Basic Data Models

In order to locally represent the preprocessed sensor data we employ two simple kinds of probabilistic models: a *threshold model* and a set of autoregressive models AR(1), AR(2), AR(3).

The threshold model is a binary model designed to estimate whether the entire segment data lies below or above a given threshold γ . The marginal likelihoods associated to these models, denoted by $m_{<\gamma}$ and $m_{>\gamma}$ resp., indicate how well the time series segment $y_{s:t}$ fits the assumptions of being below or above the threshold. For $m_{<\gamma}$ we define the improper marginal likelihood as follows:

$$P(y_{s:t} | m_{<\gamma}) = \prod_{k=s}^t p(y_k | m_{<\gamma}), \quad (1)$$

$$\text{where } p(y_k | m_{<\gamma}) = \begin{cases} 1, & \text{if } y_k < \gamma \\ p_o & \text{otherwise} \end{cases} \quad (2)$$

where $p(y_k | m_{<\gamma})$ is the probability, that a single sample y_k fits the model assumption. The parameter p_o is the probability of a simple data point y_k being an outlier w.r.t. the model. Denoting the segment length by $u = t - s$ and the number of not fitting samples by $n = |\{y_k > \gamma \mid s \leq k < t\}|$, and ignoring the constant normalization factor, we can derive the following, more compact formulas for both models:

$$P(y_{s:t} | m_{<\gamma}) = p_o^n \quad \text{and} \quad P(y_{s:t} | m_{>\gamma}) = p_o^{u-n} \quad (3)$$

As can be seen from Eq. 3, the marginal likelihood becomes smaller, the more data points are on the wrong side of the threshold.

The Autoregressive model is a special case of a general linear model $y_{s:t} = G_{s:t}^{(p)} \beta + \epsilon$, where β and ϵ denote the parameter vector and white noise respectively. The matrix of the basis vectors for the autoregressive model of order $p = 3$ is defined as follows:

$$G_{s:t}^{(3)} = \begin{pmatrix} y_{t-1} & y_{t-2} & y_{t-3} \\ y_t & y_{t-1} & y_{t-2} \\ \dots & \dots & \dots \\ y_{s-1} & y_{s-2} & y_{s-3} \end{pmatrix}.$$

Please refer to (Fearnhead, 2005), Section II and III.B for the method of likelihood calculation for this model.

Signal Preprocessing for Basic Models

The preprocessing steps are modality-specific and facilitate subsequent likelihood calculations.

Tactile signal. The tactile feedback is susceptible to strong noise and large variations within action primitives (e.g. during shaking). Thus, tactile values for each hand are summed up to yield a cumulative tactile force for each time spot. The *threshold models* are applied to this scalar time series to discriminate “object contact” from “no object contact” for each

Table 1: Overview of channel-specific models.

Sensor channel	Model	Notation
tactile sum left	threshold	m_L, m_l
tactile sum right	threshold	m_R, m_r
audio signal	AR	$m_{AR(1)}, m_{AR(2)}, m_{AR(3)}$

hand. The parameter γ specifies the threshold for recognizing hand-object contact. We denote the ‘‘object contact’’-models with capital-letter subscripts: m_L and m_R for the left and right hand respectively. The corresponding notations for the ‘‘no object contact’’-models are m_l and m_r . Note that the assignment of a ‘‘contact’’ or ‘‘no-contact’’ model by the segmentation method automatically yields an identification of the contact status during the segment.

Audio signal. Often, actions are accompanied by a typical sound, whose structure and volume remains approximately constant during the whole action primitive. Consider for example shaking an object or pouring water into a glass. Also segment boundaries are sometimes accompanied by a short, but strong change of the audio signal, e.g. placing or dropping an object.

Hence, we consider the local oscillating structure of the recorded audio signal. The signal is also subsampled and recording artifacts are removed by discarding samples whose amplitude exceeds a specified threshold. The resulting time series is logarithmized and whitened to normalize it to a given variance range w.r.t. amplitudes of individual samples. To the preprocessed data we apply the autoregressive models denoted by $m_{AR(1)}$, $m_{AR(2)}$ and $m_{AR(3)}$.

The Table 1 summarizes the association of data models to sensor channels.

Two-stage Segmentation

In our two-stage segmentation approach, we use tactile information to obtain a rough split of the sequence into subsequences of ‘‘object interaction’’ and ‘‘no object interaction’’. Subsequences that have been recognized as ‘‘object interaction’’ are analyzed in detail w.r.t. qualitative changes of the audio signal in order to refine the rough segmentation.

In the following two subsections, we describe the application of Fearnhead’s algorithm to bimanual tactile data (first segmentation step) and to audio modality (second subsegmentation step). This is based on two respective sets of models \mathcal{M} and \mathcal{M}_{sub} . Hereby \mathcal{M} consists of a mixture of product models based on the threshold model applied in the first step; \mathcal{M}_{sub} is a set of simple AR models, which is applied in the second step. The two-stage application of the segmentation procedure and the modality-specific local and bimanual models constitute the main contributions of this paper.

Segmentation Based on Tactile Modality The first step performs a rough joint analysis of the tactile signals of both hands. The analysis of bimanual data is based a mixture of four pairs of threshold models \mathcal{M} , combined in a multiplica-

tive way. Each pair corresponds to a particular contact state of the left and the right hand at once. All possible combinations of pairs define the following set $\mathcal{M} := \{m_{l_r}, m_{L_r}, m_{l_R}, m_{LR}\}$, where ‘‘no contact for both hands’’ (m_{l_r}), ‘‘contact for left hand only’’ (m_{L_r}), ‘‘contact for right hand only’’ (m_{l_R}), and ‘‘contact for both hands’’ (m_{LR}). The likelihoods of these joint models are computed as products of the individual likelihoods, e.g.:

$$P(y_{s:t} | m_{lR}) = P(y_{s:t} | m_l) \cdot P(y_{s:t} | m_R)$$

An overview of the notation can be found in the Table 2. Assignments of the four joint contact-state models to segments in a computed segmentation are illustrated in the first row of Fig. 2. Contact assignments identify parts of the time series that are directly associated with object interactions. With this approach no additional fusion is necessary for modeling of the bimanual tactile data. The contact state for both hands is determined in one pass.

Table 2: Overview of notation used for product models.

Notation	Description
m_{l_r}	no contact for both hands
m_{lR}	contact for right hand
m_{LR}	contact for left hand
m_{LR}	contact for both hands

In contrast to a pointwise application of threshold methods, Fearnhead’s method – even when used with threshold models – is not sensitive to noise which could otherwise lead to severe oversegmentation with many extremely small segments.

Sub-segmentation of Object-Contact Segments Based on Audio Modality In this subordinate segmentation step, all segments related to object interaction are further subsegmented employing the audio modality and using Fearnhead’s method once again. This time, the signal is assumed to be produced by Auto-Regressive (AR) models of order 1, 2 or 3: $\mathcal{M}_{\text{sub}} = \{m_{AR(1)}, m_{AR(2)}, m_{AR(3)}\}$. Thus the subsegmentation is formed by selecting segments that exhibit homogeneous oscillatory properties within the audio modality. The sequential application of segmentation and selection steps yields a set of segments that are characterized by constant contact topology in respect to overall hand activity as well as homogeneous characteristics of the audio signal.

Evaluation

We recorded 60 trials of the action sequence described in Sec. ‘‘Experimental Setup’’ with three subjects. This corresponds to ca. 10^5 data points and $60 \times 11 = 660$ expected change points in total. For each subject, 10 trials have been recorded with automated cue-based scheduling for ground truth and 10 trials have been manually annotated. Cue-based ground truth has been described in our previous work (Barchunova, Haschke, Franzius, & Ritter, 2011). In order

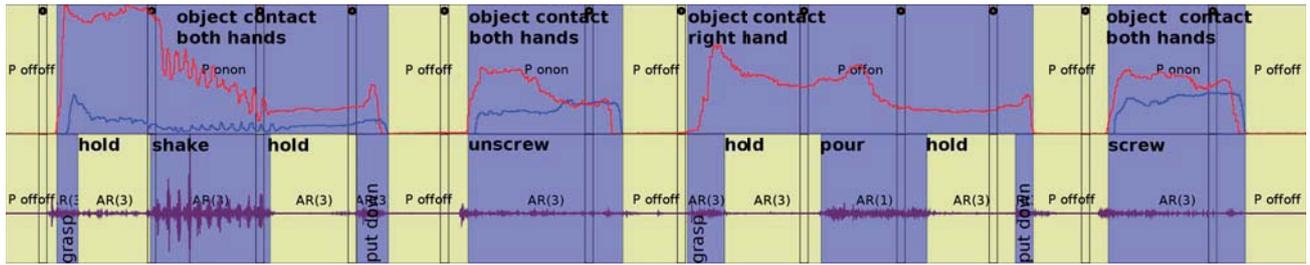


Figure 2: Hierarchical segmentation of a multimodal time series of action primitives: The first row shows the summed tactile signals of both hands and the resulting segmentation (cf. Sec. "Segmentation Based on Tactile Modality"). The segments indicated by an alternating coloring, perfectly match the underlying object-interaction structure. The second row displays the raw audio signal and the more detailed segmentation structure gained from refining object-interaction segments obtained from tactile-based segmentation (cf. Sec. "Segmentation Based on Audio Modality"). This subordinate segmentation step can correctly identify the subsequences: grasp, hold, shake, put down and pour.

to obtain the annotated ground truth for the beginning and the end of actions within the sequences, the data has been hand-labelled based on the video and audio recorded by an additional camera³. The corresponding labels have been set to match exactly the action primitives described in Sec. : *grasp+lift2*, *shake*, *hold2*, *putdown2*, *unscrew*, *grasp+lift1*, *pour*, *hold1*, *putdown1*, *screw*. In the following section we analyze the results of applying the two-stage segmentation to the constrained and unconstrained scenario.

Segmentation Quality

We assess the obtained segmentations w.r.t. the timing accuracy of the generated segmentation in both, the constrained and the unconstrained scenario. In order to assess the average error μ , the temporally closest generated change point is searched within a temporal window around the ground truth change point. The average distance between the ground truth and the generated change point measures the accuracy of the segmentation. The value μ is calculated by averaging over the trials.

The Fig. 3 (left) shows an overview of subject-specific timing deviations for all three human demonstrators *hd1*, *hd2*, *hd3* in the constrained scenario. As can be seen from the figure the average action-specific temporal error lies in the range from 0.05 seconds to 0.3 seconds for all subjects. In most cases it lies below 0.2 seconds. Furthermore, the action-specific error between different subjects varies in the range of 0.1 seconds. The variance of the error is negligible.

The Fig. 3 (right) compares the constrained and unconstrained execution scenarios. The red bars illustrate the temporal error averaged over all subjects for the constrained scenario (see left plot). The green bars present the temporal error averaged over all subjects in the unconstrained scenario. The plot does not show strong difference between the action-specific errors for both scenarios. The largest differences occur for "pour", "hold" and "putdown1".

Tests conducted with different test objects, fluids, different

³Here we have used a QuickCam Pro 9000 Analog Mono

sequences of object manipulations have showed comparable results. The assignment of product models to the bimanual tactile data in the first step has yielded 100 percent correct results. A more detailed evaluation of the segmentation procedure can be found in (Barchunova, Haschke, Grossekathoefer, et al., 2011).

Conclusions and Outlook

In this paper, we presented a novel method for unsupervised bimodal segmentation and modeling of object manipulation operations in the context of a bimanual interaction scenario. We carried out experiments with human subjects and applied the proposed method to the collected data in two different scenarios: constrained and unconstrained. The experimental evaluation has showed satisfactory results in both scenarios. In particular, the results showed invariance of the segmentation quality w.r.t. different human demonstrators and speed of execution.

The robustness and generalization ability make the method suitable for use as a building block in higher-level modeling procedures. Due to the simplicity of the two fundamental models and the modeling concepts used within our approach, the developed procedure can be easily used in a wide range of scenarios. Furthermore, the hierarchical approach to segmentation makes traditionally applied fusion unnecessary.

In our future work we seek to apply our method online within a higher-level human-machine interaction scenario.

Acknowledgments

Alexandra Barchunova gratefully acknowledges the financial support from Honda Research Institute Europe.

References

- Aggarwal, J. K., & Park, S. (2004). Human motion: Modeling and recognition of actions and interactions. In *Proceedings of the 3d data processing, visualization, and transmission, 2nd international symposium*. Washington, DC, USA: IEEE Computer Society.

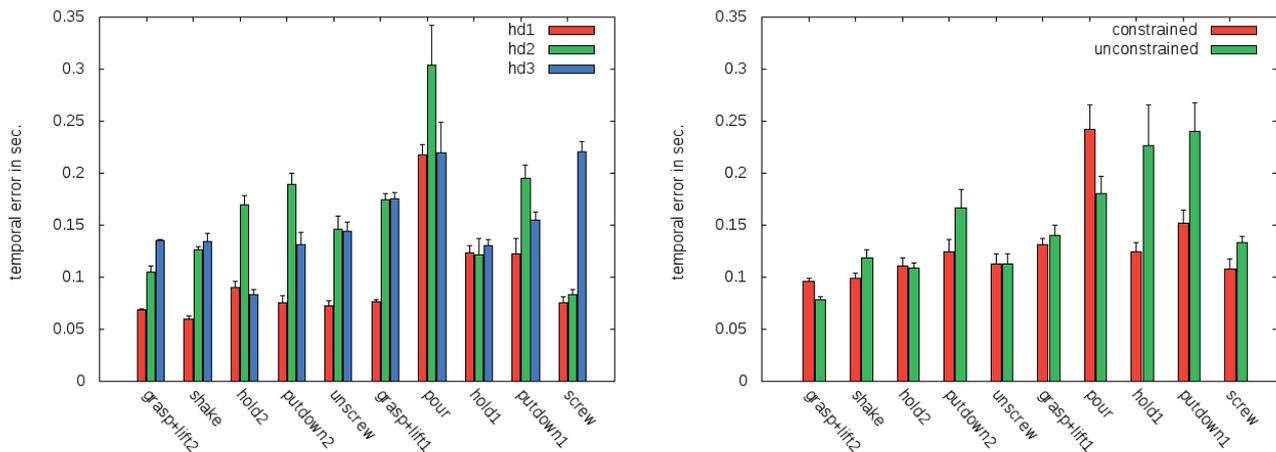


Figure 3: Action-specific timing error μ and the corresponding variance σ plotted for three human demonstrators (left subfigure); comparison between temporal error for constrained vs. unconstrained trials (right subfigure) Prior segment length parameter $\lambda = 10^{-5}$. Index 1 or 2 distinguishes between uni- and bi-manual versions action primitives respectively.

- Barchunova, A., Haschke, R., Franzius, M., & Ritter, H. (2011). Multimodal segmentation of object manipulation sequences with product models. In *Icmi*.
- Barchunova, A., Haschke, R., Grossekatheofer, U., Wachsmuth, S., Janssen, H., & Ritter, H. (2011). Unsupervised segmentation of object manipulation operations from multimodal input. In B. Hammer & T. Villmann (Eds.), *New challenges in neural computation*.
- Bernardin, K., Ogawara, K., Ikeuchi, K., & Dillmann, R. (2005). A sensor fusion approach for recognizing continuous human grasping sequences using hidden markov models. *Robotics, IEEE Transactions on*.
- Bobick, A. F., & Davis, J. W. (2001). The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23.
- Dillmann, R., Rogalla, O., Ehrenmann, M., Zöllner, R., & Bordegoni, M. (2000). Learning robot behaviour and skills based on human demonstration and advice: the machine learning paradigm. In *Robotics research*.
- Fearnhead, P. (2005). Exact bayesian curve fitting and signal segmentation. *Signal Processing*, 53.
- Grossekatheofer, U., Barchunova, A., Haschke, R., Hermann, T., Franzius, M., & Ritter, H. (2011). Learning of object manipulation operations from continuous multimodal input. In *Humanoids*.
- Kawasaki, H., Nakayama, K., & Parker, G. (2000). Teaching for multi-fingered robots based on motion intention in virtual reality. In *Iecon*.
- Krüger, V., Kragic, D., Ude, A., & Geib, C. (2007). The Meaning of Action: A Review on action recognition and mapping. *Advanced Robotics*, 21(13), 1473–1501.
- Li, C., Kulkarni, P., & Prabhakaran, B. (2006). Motion Stream Segmentation and Recognition by Classification. In *Icassp*.
- Matsuo, K., Murakami, K., Hasegawa, T., Tahara, K., & Ryo, K. (2009). Segmentation method of human manipulation task based on measurement of force imposed by a human hand on a grasped object. In *Iros*. IEEE.
- Ogris, G., Stiefmeier, T., Lukowicz, P., & Troster, G. (2008). Using a complex multi-modal on-body sensor system for activity spotting. *Wearable Computers, IEEE International Symposium*.
- Pardowitz, M., Knoop, S., Dillmann, R., & Zöllner, R. (2007). Incremental learning of tasks from user demonstrations, past experiences, and vocal comments. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 37(2), 322–332.
- Sanmohan, B., Krüger, V., & Kragic, D. (2010). Unsupervised learning of action primitives. In *Humanoid robots*.
- Takano, W., & Nakamura, Y. (2006). Humanoid robot’s autonomous acquisition of proto-symbols through motion segmentation. In *Humanoid robots*.
- Turaga, P., Chellappa, R., Subrahmanian, V., & Udrea, O. (2008). Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11), 1473–1488.
- Ward, J. A., Lukowicz, P., Troster, G., & Starner, T. E. (2006). Activity recognition of assembly tasks using body-worn microphones and accelerometers. *TPAMI*.
- Zöllner, R., Asfour, T., & Dillmann, R. (2004). Programming by demonstration: Dual-arm manipulation tasks for humanoid robots. In *Iros*.
- Zöllner, R., & Dillmann, R. (2004). Using multiple probabilistic hypothesis for programming one and two hand manipulation by demonstration. In *Iros*.

Modeling the User’s Belief about the State of a Spoken Dialog System

Klaus-Peter Engelbrecht (klaus-peter.engelbrecht@telekom.de)
Quality and Usability Lab, Telekom Innovation Laboratories, TU Berlin
Ernst-Reuter-Platz 7, 10587 Berlin, Germany

Keywords: Automatic usability evaluation; mental model, belief state.

Introduction

Spoken dialog systems (SDSs) have to respond adequately in many different situations to a multitude of different, partly misrecognized user inputs. Thus, user simulation is a valuable means to test such systems during design time. Although the user models used for the simulation are often incomplete and not always accurate, the simulated data contain much of the information found in a user test (Engelbrecht, 2012). Thus, next to reducing the effort to adapt the models to new systems, an interesting research question is how to analyze large amounts of generated data efficiently. This paper contributes to two types of analysis, namely design error detection, and prediction of perceived system quality.

Both tasks can be improved by modeling the point-of-view of the user on the dialog. One aspect of this is what the user believes to be the current system state. A wrong belief may point to concrete interface problems. On the other hand, we may be interested in how the user perceives the dialog to progress. From such data, it may be possible to derive good predictors of user judgments.

This paper presents ongoing work on this topic. We do not use a general model of cognition, but rather model this specific aspect of cognition on a conceptual level. The model is used to annotate real user interactions with an estimate of the believed system state at each dialog exchange. From this, several parameters are derived and correlated with design problems annotated in the corpus and with judgments by the users.

Belief Model

The believed system state is structured in the same way as the real system state. It consists of a set of slots (or variables) for each type of input, e.g. price range or food type. These slots are filled with values provided in the user utterances. E.g., if the utterance “I’m looking for a cheap Italian restaurant” is observed, the system would add the value “cheap” to the *price* slot, and “italian” to the *food* slot. Later, these values are used in the database query to find a matching restaurant. Contrary to the system state, the *believed* system state is not updated based on the concepts mentioned by the user, but based on the system feedback.

Recent work circling around POMDP-based, self-learning SDSs has discussed how a system may track several concurring hypotheses about the previous user inputs in a probabilistic representation of the “believed” user tasks (e.g.

Thompson et al., 2010). Although a probabilistic model would be more powerful, we use a deterministic, rule-based belief model. The reason is that users exhibit far more competencies than systems in extracting context information, which are difficult to model statistically. In addition, the parameterization and model structure are not as straightforwardly specified.

In order to exemplify the resolution level of the system, some example rules for the belief update are presented in Table 1. It can be noted that rules can refer to many, and completely different, aspects of the dialog history, which complicates the efficient probabilistic representation in a Bayesian network. In addition, processing these rules requires some annotations of the prompts, mainly with the confirmed concepts and explicit or implicit information they carry about the system state.

Table 1: Belief update rules (examples).

-
- Add concepts explicitly confirmed by the system.
 - In case affirmation of the confirmation by the user is required, and the user does not affirm or the system asks for any of the confirmed values in the next exchange, remove all confirmed values.
 - Empty slots queried by the system; however, if the system asks to repeat the value, filled the slot with an unknown value (“XXX”).
 - If the system provides no feedback, add all values of the previous user utterance, as long as the system continues consistently (e.g. not asking for one of the provided slots)
-

Use Case

In order to analyze how the belief model can support the analysis of experimental data, we use a database collected with the BoRIS restaurant information system (Möller, 2005). 40 Users (29m, 11f; $M = 29.0y$, $SD=9.7$) performed five different tasks. Three dialogs could not be used in the analysis, resulting in 197 dialogs (2001 exchanges) in the entire dataset.

Each dialog was judged on a SDS usability questionnaire. Factor analysis revealed a scale related to the overall acceptance of the system (for details, see Möller, Engelbrecht & Schleicher, 2008). In addition, log files are available, listing transcripts of each user and system turn along with speech understanding errors and task success annotations. Finally, a list of design problems was compiled and annotated at all dialog exchanges where they manifest in interaction problems.

Results

First, it is analyzed how well problematic exchanges can be predicted by the occurrence of mismatches between actual and believed system state. Intuitively, situations where the user has a wrong belief about the system state are problematic by themselves. However, we try to provide some quantification with respect to the problems annotated in the database. This is usually measured by *recall* and *precision*, where *recall* measures how many of the exchanges where a problem is annotated are also annotated with a wrong belief state. *Precision*, in turn, measures how many of the wrong belief state exchanges also have a problem annotation. We measure a *recall* of 0.50 and a *precision* of 0.66. In other words, checking the 893 exchanges where a wrong belief was annotated, half of the problematic situations are found, and 304 exchanges are analyzed in vain.

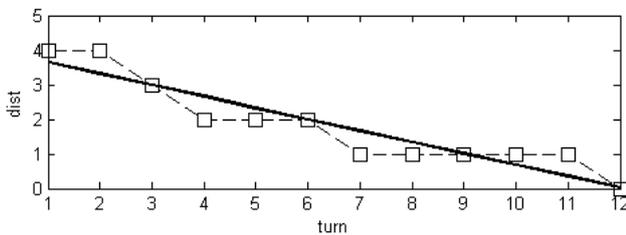


Figure 1: Distance between believed and desired system state through the 12 exchanges of an example dialog, and regression line through the points.

Next, features for the prediction of the user ratings were created. User judgments have previously been predicted from interaction data using trained classifiers (Walker, Litman, Kamm & Abella, 1997). A main problem remains to find good predictors generalizing across different systems. Analyzing the interactions from the user's perspective may be a key factor to achieve this.

First, the edit distance between the believed system state and the user goal can be determined as the number of unfilled slots plus twice the number of wrongly set slots. As illustrated in Figure 1, this distance can be specified for each exchange in a dialog. Via linear regression, the progress towards the goal can then be specified as the gradient of the regression line. Correlation analysis shows that the gradient is a fair predictor of system acceptance, compared to the standard predictors *dialog duration* and *task success* (Table 2). If we only consider the gradient over the last three exchanges, the correlation is even higher, which could be interpreted as *recency effect* (cf. Hassenzahl & Sandweg, 2004).

Table 2: Correlations of performance and judgments.

Performance	r	p
<i>task success</i>	0.26	0.00
<i>dialog duration</i>	0.31	0.00
<i>gradient(dist.), all</i>	0.26	0.00
<i>gradient(dist.), last3</i>	0.34	0.00
<i>CER</i>	0.00	0.96
<i>perceived CER</i>	-0.21	0.00

Furthermore, the perceived concept error rate (*CER*) can be calculated by comparing what the user said at each exchange to what she believes was understood. Table 2 shows that, contrary to the true *CER*, the *perceived CER* is significantly correlated with the judgments.

Conclusion

This paper showed, using an example SDS, that modeling the belief users have about the system state over the course of a dialog can provide valuable information for data analysis. Differences in the believed and desired system state (vaguely) hinted to system design errors. In the future, more qualified indicators may be derived from the belief annotations. Furthermore, new parameters for the prediction of user judgments were derived and showed correlations with the judgments in the range of *task success* and *dialog duration*. Subsequent research will show if the new parameters are independent from previous ones and thus useful as *additional* predictors.

Unfortunately, as many different parameters and complex relations between the dialog acts of user and system need to be exploited to update the believed system state, no sound probabilistic model could be presented at this stage. In addition, the generalization of the model to other SDSs has to be tested. Finally, other knowledge users collect about the system during a dialog could be tracked to analyze the data more comprehensively and run user simulations with the models. All this will be dealt with in future work.

References

- Engelbrecht, K.-P. (2012). Estimating Spoken Dialog System Quality with User Models. TU Berlin.
- Evanini, K., Hunter, P., Liscombe, J., Suendermann, D., Dayanidhi, K. & Pieraccini, R. (2008). Caller Experience: A Method for Evaluating Dialog Systems and its Automatic Prediction. *Proc. of SLT* (pp. 129-132). Goa, India.
- Hassenzahl, M., Sandweg, N. (2004). From Mental Effort to Perceived Usability: Transforming Experiences into Summary Assessments. *Proc. of CHI*, Vienna, Austria.
- Hastie, H. W., Prasad, R., Walker, M. (2002). Automatic Evaluation: Using a Date Dialogue Act Tagger for User Satisfaction and Task Completion Prediction. *Proc. of LREC 2002* (pp. 641-648), Las Palmas de Gran Canaria.
- Möller, S. (2005). Quality of Telephone-based Spoken Dialog Systems. New York, USA: Springer.
- Möller, S., Engelbrecht, K.-P., Schleicher, R. (2008). Predicting the Quality and Usability of Spoken Dialogue Services. *Speech Communication*, 50, 730-744.
- Thomson, B., Jurcicek, F., Gašić, M., Keizer, S., Mairesse, F., Yu, K., & Young, S. (2010). Parameter learning for POMDP spoken dialogue models. *Proc. of SLT* (pp. 271-276), Berkeley, California.
- Walker, M., Litman, D., Kamm, C., Abella, A. (1997). PARADISE: A Framework for Evaluating Spoken Dialogue Agents. *Proc. of ACL/EACL* (pp. 271-280), Madrid, Spain.

A four factor model of landmark salience – A new approach

Florian Röser (florian.roeser@psychol.uni-giessen.de)
Antje Krumnack (antje.krumnack@psychol.uni-giessen.de)
Kai Hamburger (kai.hamburger@psychol.uni-giessen.de)
Markus Knauff (markus.knauff@psychol.uni-giessen.de)

Justus Liebig University Giessen, Department of Psychology,
Experimental Psychology and Cognitive Science
Otto-Behaghel-Strasse 10 F
35394 Giessen, Germany

Abstract

Until today there exist few theoretical assumptions about the concept of landmark salience. They could be divided into two fields: a more physical view (inherent aspects of the landmarks) and a more cognitive/personal view (the validation of the specific landmark from the individual cognitive features). We here combine these two aspects and present first empirical evidence for the interdependence of visibility and structural salience.

Keywords: wayfinding; landmarks; perceptual, cognitive, structural salience; visibility

Introduction

What is a landmark and how can a landmark be defined? Today there exist several definitions of landmarks (e.g., Lynch, 1960; Presson & Montello, 1988), additionally models of landmark salience have been put up (e.g., Klippel & Winter, 2005; Caduff & Timpf, 2008).

While Lynch (1960) assumes objects to have inherent physical features that make them a landmark, Presson and Montello (1988) emphasized the importance of visual contrast of an object to its immediate surrounding. Thus, visual aspects seem to play a major role in landmark and wayfinding research. Within this context Caduff and Timpf (2008) proposed the importance of “relatively distinct, prominent or obvious features compared to other features” (p. 250). This leads to a competition between different objects to be chosen as landmarks. In other words, they need to draw our attention (extrinsically as well as intrinsically). Such competition will serve as basis for a) comparing different salience concepts and b) establishing our own salience model based on the previous concepts, empirical findings, and modeling.

Our first assumption is based on the approach by Gärling, Böök, and Lindberg (1986). It is defined in more detail by Caduff and Timpf (2008) and means that there is a trilateral relationship between the observer, the object (that is potential a landmark) and the environment (figure 1). This implies that the object cannot be assessed without the context.

Caduff and Timpf’s (2008) model includes the three concepts of salience: perceptual (the bottom-up perception),

cognitive (top-down factor; wayfinders’ experience and knowledge), and contextual (measure of attention that the wayfinder can render). Furthermore, they focus on the personal and cognitive aspects of a wayfinder in the context of wayfinding and landmarks as highlighted in the trilateral relationship (figure 1).

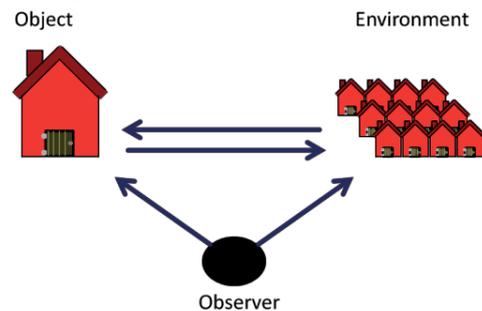


Figure 1: Observer, object, and environment have a trilateral relationship (based on Caduff & Timpf, 2008).

Sorrows and Hirtle (1999) proposed a concept which concentrates more on the physical aspects of the landmarks: visual (visual characteristics of the landmark), cognitive (meaning or prototypicality), and structural (location in space) salience. Based on this landmark salience concept, Klippel and Winter (2005) proposed a mathematical model and amended it with the concept of visibility (Winter, 2003). Each quality in their model is expressed by a normed measure of salience (with values in the interval $[0, 1] \subset \mathbb{R}$). These individual measures are combined to a joint salience of a landmark in a weighted sum (formula 1). This joint salience is moderated by the visibility (formula 2):

$$S_O = W_v S_v + W_s S_s + W_u S_u \quad \text{with } W_v + W_s + W_u = 1 \quad (1)$$

$$S_t = V * S_O = VW_v S_v + VW_s S_s + VW_u S_u \quad (2)$$

S_O = joint salience; S_v = visual salience;
 S_s = semantic salience; S_u = structural salience,
 W_v, W_s, W_u = weighting factors,
 S_t = total salience; V = visibility

We use this formula as a starting point but adapt the definitions of the components. In our model we combine cognitive and personal aspects similar to Caduff and Timpf (2008) with physical aspects similar to Sorrows and Hirtle (1999). We think that the following four major aspects constitute the salience of a landmark:

1. perceptual salience (ρ),
2. cognitive salience (c),
3. structural salience (s), and
4. visibility (v).

Perceptual salience (ρ)

We define the perceptual salience as the physical aspects of the object (color, shape, texture, orientation, height, weight; these are not the only ones, see Hamburger & Röser, 2011). One inherent aspect is the contrast to the surroundings as described by Itti and Koch (2001). Objects need to be highly noticeable in order to pop out from the surroundings (Presson & Montello, 1988; Janzen & van Turenhout, 2004).

Therefore, this salience type should reflect the setting at an intersection. A good landmark at one intersection can be a bad landmark at another intersection. A red house, for example, generally has a high perceptual salience but if it is located in a series of red houses, or if all houses at an intersection are red, then it has a low or insignificant perceptual salience.

We assume that there is an absolute perceptual salience, but that this property is moderated by the context to judge the suitability of a landmark. Therefore, we need a measure of how much the salience level of a landmark stands out from the salience values of other landmarks. To achieve this, we compare how much the perceptual salience differs from the average salience values at an intersection and only consider salience values to be relevant which are higher than the average at the intersection. To achieve this, we subtract the average salience value of the other landmarks at the intersection from the salience value of the landmark in question. We use a maximum function to ensure the resulting value is at least zero. This looks as follows for an landmark A with an absolute perceptual salience s_{pA} at an intersection i with a set of landmarks (L) Λ :

$$s_{pA_i} = \max\left(s_{pA} - \frac{\sum_{L \in \Lambda \setminus A} s_{pL}}{|\Lambda \setminus A|}, 0\right), \quad (3)$$

where s_{pA_i} is the intersection specific perceptual salience for landmark A .

Cognitive salience (c)

The cognitive salience is based on the personal, intellectual, and cultural background of the wayfinder. Again, the manifestation of this can only be considered within the direct context of the landmark (see above for the similar description of the perceptual salience). Imagine there is a gas station at an intersection. For a common car driver it could serve as a landmark with a high cognitive meaning.

But, if there are two or three gas stations at this same place every single one has a low or insignificant cognitive salience.

To express this, we used the same formula as for the perceptual salience. So for a landmark A with an absolute cognitive salience s_{cA} at an intersection i with a set of landmarks (L) Λ we get:

$$s_{cA_i} = \max\left(s_{cA} - \frac{\sum_{L \in \Lambda \setminus A} s_{cL}}{|\Lambda \setminus A|}, 0\right), \quad (4)$$

where s_{cA_i} is the intersection specific cognitive salience for landmark A .

If no object perceptually or cognitively contrasts the other objects at an intersection (that means if all objects are equal with regard to perceptual or cognitive concepts) then by definition the perceptual and cognitive salience values are zero.

Structural salience (s)

We define the structural salience as a local salience (Klippel & Winter, 2003) which reflects the position of the landmark at an intersection, and thereby the structure of the intersection where it can be found. We assume that structural salience has the same distribution for every four-way, right-angled intersection (figure 2). Positions with a high structural salience can be viewed as places where people prefer to look for landmarks, which tend to be in the direction of the turn (see Röser, Hamburger, Krumnack & Knauff, in press). Or, in other words, the structural salience is based on the attention since it will be on the direction of the path.

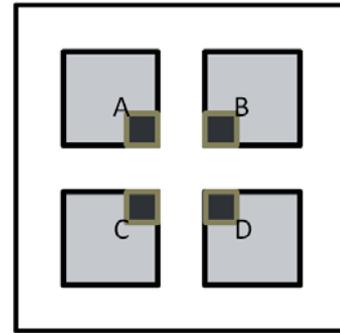


Figure 2: Four positions: two before and two after the intersection. Two directions: in direction of the turn or opposite to the direction of the turn (left, right).

Visibility (v)

Here, visibility is defined as a viewpoint based visibility which is based on the position at which the participant has to decide in which direction to move on (Röser, Hamburger, Krumnack & Knauff, in press). This is in contrast to the advanced visibility by Winter (2003).

First of all, we assume that there is a visibility threshold for the perceptual and cognitive salience: if the visibility is so low that you cannot recognize the quality of the landmark

that induces the salience, then that type of salience does not contribute to the total salience of the landmark. If the visibility is high enough for the observer to recognize the quality of the landmark that induces the salience, then that type of salience is not limited by visibility. For example, consider the identification of a train station. First, the wayfinder will only see a large building but there will be one point at which he could identify it as a train station even if he cannot see it clearly. On the other hand, if there is a red house in a haze so that the wayfinder could not identify it (or perceive it) than it is not usable as a landmark.

Therefore we define specific visibility values v_p and v_c to be multiplied by the perceptual and cognitive salience:

$$v_p = \begin{cases} 1, & \text{if perceptual sal. of the landmark is noticeable} \\ 0, & \text{otherwise} \end{cases},$$

and accordingly

$$v_c = \begin{cases} 1, & \text{if cognitive sal. of the landmark is noticeable} \\ 0, & \text{otherwise} \end{cases}.$$

However, visibility does not have this all or nothing effect on structural salience.

Now we have all the necessary components for defining our model. Substituting our definitions in formula (2) we get:

$$s_t = v_p w_p s_{pA_i} + v_c w_c s_{cA_i} + v w_s s_s. \quad (5)$$

Experiments

In the following experiments we will examine the influence of the visibility on the structural salience by eliminating the influence of the perceptual and cognitive salience (thus, we only investigate two factors). For this we use the combination of four colors and four shapes as landmarks. We assume that these landmarks have an equal perceptual and cognitive characteristic. By definition (formula 3 and 4), we assume that these aspects do not influence the results, leading to the following formula:

$$s_t = v_p w_p * 0 + v_c w_c * 0 + v w_s s_s = v w_s s_s. \quad (6)$$

For a variation of the influence of the visibility we used different perspectives within our virtual environment SQUARELAND (Hamburger & Knauff, 2011): an allocentric and egocentric point of view (figure 3). In the allocentric condition the visibility is identical for all possible landmark positions at an intersection, while in the egocentric condition different visibilities emerge (e.g., amount of occlusion), depending on the position of the landmark at the intersection.

Landmark material

As landmarks we used four shapes (triangle, square, hexagon, and circle) that could have one of four colors

(yellow, green, blue, and red), resulting in 16 landmarks. Each of the 16 landmarks was randomly distributed to the four positions at the 16 intersections in the maze and at each intersection no shape or color were presented twice. Depending on the direction of the turn (see figures 3 and 6), each of the shapes and colors were presented four times at each position.

Experiment 1 – Allocentric perspective

Methods

A total of 26 participants (18 females; 21 students) completed the online questionnaire. Participant's mean age was 22.88 years (range: 19-38). All participants provided informed written consent. The students received course credit for participation.

Material

A 2D maze consisting of 5 X 8 squares and orthogonal angles at each intersection was designed for this experiment (figure 2). For each decision there was a new map (image) with the route visualized up to the current position and decision point (figure 3). The maze with the paths and intersections was created in Word2007 (Microsoft Office) and LimeSurvey 1.85 was used to run the online questionnaire and for data recording.

Procedure

In the online questionnaire participants were presented with a short instruction to learn the given route with a map (16 intersections). Subsequently, they saw a short cover story (“imagine you must describe the learned route to someone who is unfamiliar with this route but needs to find the goal location of this route”). Then they were shown the path from the start point to the first intersection (allocentric perspective), and had to answer the question “which of the following descriptions (e.g., at the green hexagon to the right) appears to be most convenient for you”. This procedure was repeated for all intersections.

Results

Results for colors, shapes, and position at the intersections are visualized in figure 4. They revealed no differences between the four shapes ($\chi^2(3)=0.201$, $p=.976$) and the four colors ($\chi^2(3)=.221$, $p=.974$). Clear preferences of the participants for landmark positions were obtained: on the right hand side of an intersection in case of a right turn (with 91.35%) and the left side in case of a left turn (88.91%). Such obvious preferences made any statistical analysis needless. Furthermore, for landmark positions in the direction of the turn, positions before the turn –the object has to be passed before the turn is executed– are selected 4.1 times more often than the position after (behind) the intersection –where the object is not physically passed.

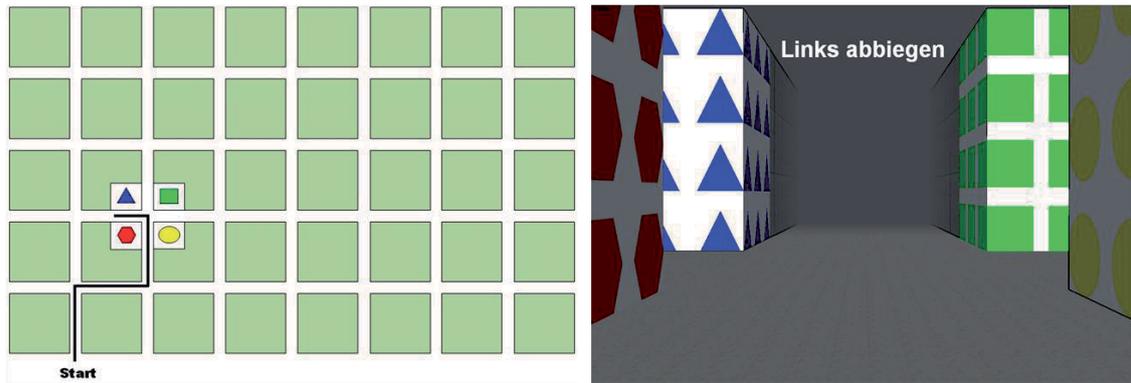


Figure 3: Allocentric view of the maze (SQUARELAND) on the left. Egocentric perspective on the right (instruction: “Links abbiegen” = “Left turn”). Landmarks are presented at the walls.

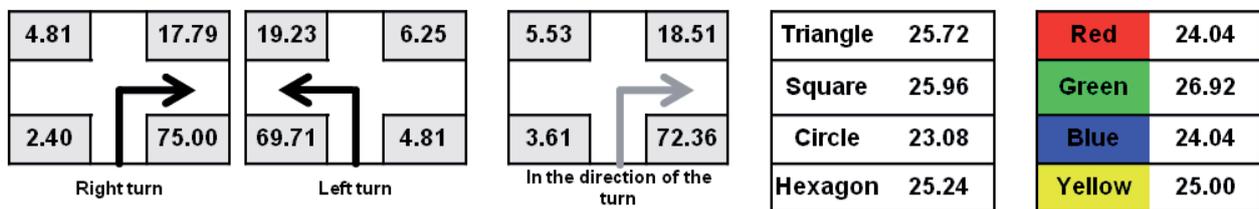


Figure 4: Relative choices of the single landmark positions over all intersections, shapes, and colors in the allocentric perspective (left). Relative choices for single shapes and colors over all intersections and landmark positions (right).

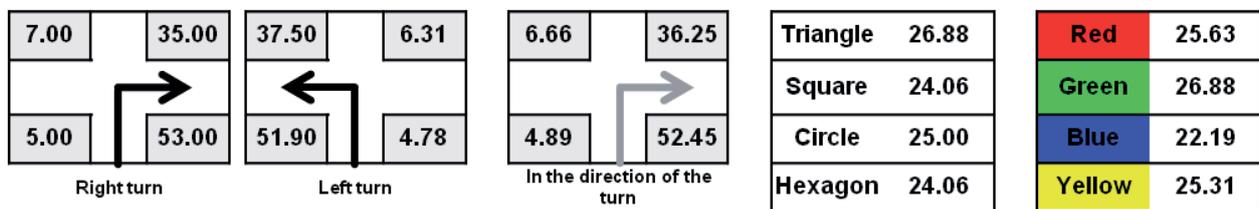


Figure 5: Relative choices of the single landmark positions over all intersections, shapes, and colors in the egocentric perspective (left). Relative choices for single shapes and colors over all intersections and landmark positions (right).

Discussion

For the allocentric perspective the visibility is the same for all intersections and positions and could therefore be ignored. Thus, we here only measured the inherent saliences of the landmarks, namely the structural salience (the other saliences assumed zero, see above).

In summary, we could determine that the position in the direction of the turn, before the intersection is the optimal one for wayfinding/route descriptions. This is in line with the assumption of Klippel and Winter (2005).

Experiment 2 – Egocentric perspective

We re-examined the structural salience of Experiment 1 with an egocentric perspective within a virtual maze. Again, we had a learning phase in which the participants had to learn a route direction and decide/imagine at which position the landmark could/should ideally be located.

Methods

A total of 20 students from the University of Giessen (11 females) participated in this experiment. They had a mean age of 22.9 years (range 19-29). They all had normal or corrected-to-normal visual acuity, provided informed written consent and received course credit for participation.

Material

For this experiment we used the 3D version of the virtual environment SQUARELAND (Hamburger & Knauff, 2011) as described above. The walls and floor were light and dark gray and a neutral gray haze was implemented in the background, so that participants could only see the next intersection but no additional landmark information.

A video led the participants passively along the path through the maze with 16 intersections. The route and positions of the landmarks (combinations of colors and shapes) were the same as in the allocentric experiment above; figure 3). The route direction and the video were presented by a Panasonic PT-F100NT projector. The full

image subtended 67 deg in height and 85 deg in width of the observers' visual field. Superlab 4.0 (Cedrus Corporation 1991-2006) was used for running the experiment and for data recording (for more details see Röser et al., in press).

Procedure

The procedure was the same as for the allocentric experiment with the difference that the participants now saw a video (trail) (egocentric perspective) from the start point to the first intersection (where they had to decide which landmark they would prefer). Here the direction instruction was given in midair (figure 3). After each trial, the video started over until the next intersection was reached where participants again indicated the preferred landmark.

Results

The results for the colors, shapes, and positions at the intersections are visualized in figure 5. No differences between the four shapes ($\chi^2(3)=0.212, p=.976$) and the four colors ($\chi^2(3)=.477, p=.924$) were found. The participants preferred the positions in the direction of the turn with 88.70%.

Looking at the absolute frequency of the specific positions mentioned, we tested them for uniform distribution. Here we obtained a significant difference ($\chi^2(3)=209, p<.001$). If we take the relative frequency for how often each position was mentioned across all trials by a single participant and across all participants, we calculated a one-factorial analysis of variance (ANOVA). This analysis revealed a significant difference ($F(3)=9.72, p=.003$). The post hoc t-test revealed the following for the different positions:

Position	T-values	P-values
A ↔ B	-3.432	.003
A ↔ C	0.698	.494
A ↔ D	-4.276	<.001
B ↔ C	3.617	.002
B ↔ D	-0.941	.358
C ↔ D	-4.753	<.001

- A = after intersection, opposite the direction of the turn
- B = after intersection, in the direction of the turn
- C = before intersection, opposite the direction of the turn
- D = before intersection, in the direction of the turn (compare to figure 6)

Discussion

There is no difference between the position before and after the intersection (independently, in the direction of the turn or opposite). This contradicts the general assumption that people prefer the position before an intersection (e.g. Klippel & Winter, 2005).

Since in this experiment the visual and semantic salience may also be assumed to be zero (see above), thus, we only measured the influence of the structural salience moderated by the visibility. The straightforward follow-up question

then is: What is the influence of these factors and can we predict the results of the egocentric experiment with the values from the allocentric experiment and the visibility?

Modeling visibility measure

To measure the visibility we first regard the visible parts of the landmarks at each decision point that is which proportion of the facades facing the intersection is visible (figures 6).

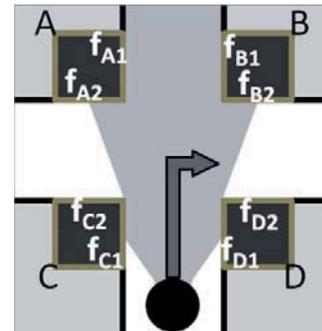


Figure 6: Intersection with the decision point and the visible parts of the landmarks from this position. The gray highlighted area on the path is the field of view. The arrow gives the direction of the turn (summarized for a left and a right turn).

Based on this we come to the following specific visibilities for the facades of the landmark: $f_{A1} = f_{B1} = 1$; $f_{A2} = f_{B2} = 0.48$; $f_{C1} = f_{D1} = 0.48$; and $f_{C2} = f_{D2} = 0$.

To calculate the visibility for one landmark we must average the visibilities of its two facades:

$$v = \frac{v_{f1} + v_{f2}}{2} \quad (7)$$

This results in the following visibility values:

$$\begin{aligned} v_A &= 0.74, \\ v_B &= 0.74, \\ v_C &= 0.24, \\ v_D &= 0.24. \end{aligned}$$

With formula (6), the visibility and the results of the allocentric Experiment 1 can be used to predict the results for the egocentric Experiment 2 (figure 7).

To do so, we multiply the results of Experiment 1 with the specific visibility. However, these calculated values cannot be compared directly to the results of Experiment 2 for the following reason: the measured structural salience of a single landmark position depends on the measured salience of the other landmark positions. Due to the fact that the participants always had to choose one position, the sum of all measured frequencies in our setups is always 1. The sum of our calculated values is less than 1. To adjust the prediction we divided each calculated value by the sum of all calculated values so that the resulting values add up to 1 (compare to figure 7). This operation does not change the

ratio of the values for the different positions. The numbers obtained from these calculations indicate that the results from the allocentric experiment and the visibility are good predictors for the results of the egocentric experiment. Consequently, we may summarize that the interdependence between the visibility and the structural salience defined by Klippel and Winter (2005) and our new model could be empirically confirmed.

Conclusion and further experiments

Visibility seems to have the effect predicted by Klippel and Winter (2005) on structural salience. At the start we had five questions:

- What determines the salience of a landmark?
- What determines the distribution of landmarks chosen?
- What is the influence of the surroundings on the above issues? Is this fully expressed in structural salience?

- Is there an interaction between cognitive and structural salience or is the cognitive salience just influenced by the surrounding?
- If there is an interaction, what does it look like?

The first three questions can be answered with the model above. Currently we investigate the combination of the structural and cognitive salience. First results show that there is an interference between the four positions at an intersection and the influence of the cognitive characteristics. We hope to define this interaction and the weight factors in our formula (5) with this and further experiments.

With these experiments and model we found a first empirical answer to the question which position should be used for a landmark (especially in route descriptions or navigation systems) to be in accordance with human spatial abilities. The remaining two saliences and their influence on human wayfinding will be subject to further experiments.

	Results of the allocentric experiment	Multiplicated with the specific visibility	Predicted values (divided by 0.36)	Results of the egocentric experiment
After the intersection, opposite the direction of the turn (A)	0.06	0.04	0.12	0.07
After the intersection, in the direction of the turn (B)	0.18	0.13	0.37	0.36
Before the intersection, opposite the direction of the turn (C)	0.04	0.01	0.03	0.05
Before the intersection, in the direction of the turn (D)	0.73	0.18	0.48	0.52
Sum	1	0.36	1	1

Figure 7: Calculation for the prediction of salience values for the egocentric experiment, based on the results of the allocentric experiment and the defined visibility.

References

- Caduff, D., & Timpf, S. (2008). On the assessment of landmark salience for human navigation. *Cognitive Processing, 9*, 249-267.
- Gärbling, T., Bööck, A., & Lindberg, E., (1986). Spatial Orientation and Wayfinding in the Designed Environment: A Conceptual Analysis and Some Suggestions For Postoccupancy Evaluation. *Journal of Architecture, Plans and Resources, 3*, 55-64.
- Hamburger, K., & Knauff, M. (2011). SQUARELAND: A virtual environment for investigating cognitive processes in human wayfinding. *PsychNology Journal, 9*, 137-163.
- Hamburger, K., & Röser, F. (2011). The meaning of Gestalt for human wayfinding – How much does it cost to switch modalities? *Gestalt Theory, 33*, 363-382.
- Itti, L., & Koch, C. (2001). Computational modelling of visual attention. *Nature Reviews Neuroscience, 2*, 194-203.
- Janzen, G., & van Turenout, M. (2004). Selective neural representation of objects relevant for navigation. *Nature Neuroscience, 7*, 673-677.
- Klippel, A., & Winter, S. (2005). Structural salience of landmarks for route directions. In A. G. Cohn & D. M. Mark (Eds.), *Spatial Information Theory. International Conference COSIT* (pp. 347-362). Berlin: Springer.
- Lynch, K. (1960). *The image of the city*. Cambridge, MA: MIT Press.
- Presson, C. C., & Montello, D. R. (1988). Points of reference in spatial cognition: Stalking the elusive landmark. *British Journal of Developmental Psychology, 6*, 378-381.
- Röser, F., Hamburger, K., & Knauff, M. (2011). The Giessen virtual environment laboratory: human wayfinding and landmark salience. *Cognitive Processing, 12*, 209-214.
- Röser, F., Hamburger, K., Krumnack, A., & Knauff, M. (in press). The structural salience of landmarks: Results from an online study and a virtual environment experiment. *Journal of Spatial Science*.
- Sorrows, M. E., & Hirtle, S. C. (1999). The nature of landmarks for real and electronic spaces. In: C. Freksa & D. M. Mark (Eds). *Spatial information theory: cognitive and computational foundations of geographic information science, international conference COSIT* (pp. 37-50). Stade: Springer.
- Winter, S. (2003). Route adaptive selection of salient features. In: W. Kuhn, M. F. Worboys, & S. Timpf (Eds.), *COSIT 2003, LNCS 2825* (pp. 349-361). Berlin: Springer.

Modeling the Effects of Base-rates on Cyber Threat Detection Performance

Varun Dutt (varundutt@cmu.edu)
Young-Suk Ahn (ysahn@altenia.com)
Noam Ben-Asher (noamba@andrew.cmu.edu)
Cleotilde Gonzalez (coty@cmu.edu)

Dynamic Decision Making Laboratory, Department of Social and Decision Sciences
Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA

Abstract

Cyber attacks cause major disruptions of online operations, and might lead to data and revenue loss. Thus, appropriately training security analysts, human decision makers who are in charge of protecting the infrastructure of a corporate network from cyber attacks, on different frequencies of cyber threats (base-rates) is indispensable to improving their on-job performance. However, little is currently known about how training analysts on different cyber attacks, that differ in the base-rate of cyber-threats, affects their on-job performance in a highly dynamic environment, while confronting novel transfer conditions. We report a laboratory experiment where human participants are trained on two different cyber-threat base-rates, high and low, and are transferred to an intermediate base-rate level of threats. The experiment helps us to develop an understanding of the situational attributes that participants attend to during their detection of cyber-threats. A linear model that is based upon participants' attended attributes and calibrated to the two base-rates during training does well to capture the performance during transfer. We use the calibrated model to generate predictions in novel real-world transfer conditions that contain a low cyber-threat base-rate and a shorter training period.

Keywords: cyber-threat; linear model; security analyst; training; transfer; base-rate.

Introduction

Cyber attacks, i.e., the disruption of computers' normal functioning and the loss of sensitive information in a network through malicious network events (cyber-threats), are becoming widespread. With "Anonymous" and other threats to corporate and national security, guarding against cyber attacks is becoming a significant part of IT governance, especially because most government agencies and private companies have moved to online systems (Sideman, 2011). Recently, President Barack Obama declared that the "cyber-threat is one of the most serious economic and national security challenges we face as a nation" (2011). According to his office, the nation's cyber-security strategy is twofold: (1) to improve our resilience to cyber incidents; and (2) to reduce the cyber-threat. To meet these goals, the role of a security analyst (called the "analyst" hereafter), a human decision maker who is in charge of protecting the online infrastructure of a corporate

network from random or organized cyber attacks, is indispensable (Jajodia, Liu, Swarup, & Wang, 2010).

Given that the threat of cyber attacks is growing, there is an urgent need to emphasize training programs for analysts that will acquaint them with different kinds of attacks. For example, the U.S. Department of Homeland Security (DHS) has recently started offering a weeklong training program to help analysts learn how to deal with intrusions into their computer networks (Zakaria, 2011). In this training program, the DHS uses a scenario that contains industrial espionage: a fictitious company ACME has built a new chemical product and another company Barney Advanced Domestic (BAD) Chemicals tries to steal its "secret sauce" and disrupt operations to put ACME out of business (Zakaria, 2011).

Although training programs like the one by DHS are an important step towards improving cyber-threat detection, there is little literature on how training analysts on different kinds of cyber attacks will influence their on-job performance (Dutt, Ahn, & Gonzalez, 2011). One aspect of analyst training is base-rate: In the real world, cyber threats are likely to occur with a low base-rate and given this rare occurrence, analysts might underweight them by relying on their personal experience (Hertwig & Erev, 2009). Another aspect of analysts' training is length. Lengthy training is likely to benefit analysts, but also becomes costly in resources and time (Kanellis, 2006). In this regard, little is known about how the training period might influence analysts' performance. The current documentation about analysts' decisions is sensitive information that is classified, as it could be exploited by attackers (D'Amico et al., 2005). Thus, in the absence of real human data, literature has proposed a simulation approach towards evaluating the effects of training manipulations on performance at transfer (Dutt, Ahn, & Gonzalez, 2011). For example, Dutt, Ahn, and Gonzalez (2011) have proposed a computational model based upon the Instance-Based Learning Theory (IBLT; and, "IBL model" hereafter; Gonzalez & Dutt, 2011), to generate predictions about the effects of training simulated analysts with different base-rates on their transfer performance. They pre-populated the model's memory with experiences of a threat-prone base-rate (90% threats and 10% non-threats) and a nonthreat-prone base-rate (10% threats and 90% non-threats).

Is threat	Alert	Description
<input type="checkbox"/>	No Alert	A user outside the company sends a packet #B to webservice, but #B fails.
<input checked="" type="checkbox"/>	#B has signature compromising the webservice	A user inside the company sends a packet #B to the webservice, and #B succeeds.
<input type="checkbox"/>	No Alert	A user inside the company sends a packet #B to the webservice, and #B fails.
<input checked="" type="checkbox"/>	No Alert	A user outside the company sends a packet #B to webservice, and #B succeeds.
<input type="checkbox"/>	#B has signature compromising the webservice	A user inside the company sends a packet #B to the webservice, and #B succeeds.

Figure 1: A typical trial showing events' description and accompanying alert information. Participants marked an event as a threat by checking the "Is threat" box against the event's description.

snapshot of a trial with 5 events (some with alerts and some without). The participants' main goal in the task was to

The model with a threat-prone memory possessed a greater hit rate and a smaller false-alarm rate (i.e., greater accuracy) compared with the model

with a nonthreat-prone memory. These results have been replicated by Dutt and Gonzalez (in press). Thus, when human analysts are exposed to higher base-rates during their training, they are more likely to perform better at transfer. However, these model predictions are limited by the lack of empirical validity from real human observations.

In this paper, we build upon existing literature and report a laboratory experiment that aims to empirically evaluate analysts' cyber-threat detection accuracy when they are trained on different base-rates before transferring to novel conditions. Furthermore, we evaluate the attributes to which participants attend during their training with self-reported strategies. Using this information, we propose a linear model that classifies cyber events as threats and non-threats. The proposed model fits the existing data well. Finally, we use this model to generate predictions in novel real-world transfer conditions that contain a low cyber-threat base-rate and a shorter training period. The low base-rate is likely to be representative of situations encountered in the real world, where cyber-threats rarely occur in a certain period of time (Jajodia et al., 2010). Shorter training periods might be plausible, given the resource and time costs imposed by lengthy training sessions.

Experiment: Effects of Cyber-Threat Base-rates on Threat Detection Performance

We report an experiment where we train participants in two different conditions, high and low, that differ in the proportion of cyber-threats present during training. We evaluate the effects of their training with base-rates on their transfer performance. Our main motivation is to find the attributes to which participants attend most. According to prediction from the IBL model proposed by Dutt, Ahn, and Gonzalez (2011), we expect better performance (i.e., greater hit rates and smaller false-alarm rates) from participants who are trained in conditions involving higher base-rates. That is because higher base-rates provide participants with more frequent opportunities to formalize and test different hypotheses regarding what defines a threat.

The experiment involved two between-subjects conditions, low (N=20) and high (N=18), that differed in the proportion of cyber-threats presented in a trial. Both conditions contained 10 training trials followed immediately by a transfer trial. Each trial contained 25 network events that were presented to participants sequentially one at a time. An event included a description and might be accompanied with an alert. The alerts were generated by an intrusion detection system (IDS) that might indicate whether these events were threats or not. The IDS systems generated both false-positives and false-negatives. Figure 1 shows a

correctly classify each event as a threat or a non-threat by checking or unchecking the corresponding "Is threat" box for each event. A new event appeared in the window after every four seconds. Participants could go back and check/uncheck any previously presented event during the duration of the trial, but not after a trial had ended.

During training in the low base-rate condition, 12% of events (=3) were actual threats in each training trial. For the high base-rate condition, 52% of events (=13) were threats in each training trials. A threat was defined as any event that by its description was: (1) initiated by a user outside the company; and (2) against which there was an alert generated. However, participants were not told this definition and they were expected to discover it with practice. In both conditions, a single transfer trial was presented and 32% of events were threats (=8). Thus, the base-rate in the transfer trial was in between the low and high conditions (participants were not told about any base-rates). Participants' performance was evaluated in terms of hit and false-alarm rates at the end of each trial.

Instructions informed participants about how they would be paid based upon their performance. After participants read the instructions, they started the experiment with the first training trial. At the end of each training trial, they were asked to write down the strategy that guided their events' classification. After participants submitted their written explanations, they were informed of the number of hits, misses, false-alarms, and correct-rejections they made in the last trial; along with their current and total earnings based upon performance. However, they were not shown which exact events in the last trial were actual threats and non-threats. Participants were compensated with \$5 as base payment. In addition, participants earned 1 cent for each threat and non-threat correctly classified and lost 1 cent for each threat and non-threat incorrectly classified during training. During transfer, participants earned 3 cents for each threat and non-threat correctly classified and lost 3 cents for each threat and non-threat incorrectly classified. After participants had completed their experiment, they were paid and thanked for their time.

Results

We expected superior performance (i.e., a greater hit-rate and a smaller false-alarm rate) in the high condition compared with the low condition. Figure 2 shows the aggregated hit and false-alarm

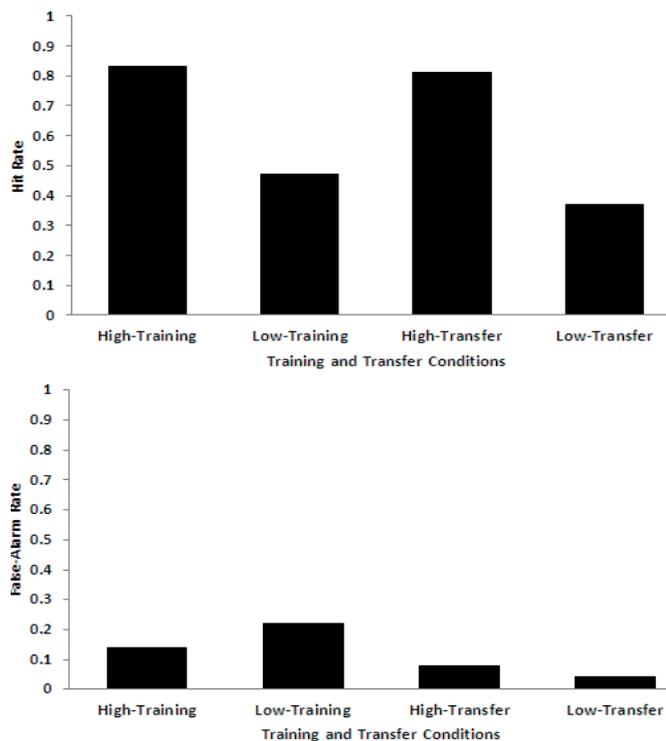


Figure 2: The human hit rate and false-alarm rate in the low and high conditions during training and transfer.

rates in both conditions during training and at transfer. As can be seen in Figure 2, during training, the average hit rate in the high condition (83%) was significantly greater than that in the low condition (47%), $t(36) = -3.89, p < .001$. The same relationship existed for the hit rate at transfer between the low condition (37%) and the high condition (81%), $t(36) = -4.95, p < .001$. Although false-alarm rates generally appeared to be greater in the low condition compared with the high condition, these differences were insignificant, both during training ($t(36) = 1.74, ns$) and at transfer ($t(36) = -1.23, ns$). Overall, these results are in agreement with our expectation of superior performance in the high condition.

Attention to Situational Attributes

In order to gain deeper understanding of the overall performance in the detection task, we analyzed the explanations that participants provided about their classifications at the end of each trial. Although there was diversity present in their explanations across trials, five main categories emerged in each condition for which a majority of explanations could be categorized into (45% and 35% in the low and high conditions). Table 1 provides a description of these categories with statistical differences between the two conditions. We believe participants' explanations under different categories were based upon the descriptions of events and corresponding alerts that were presented to them during training and transfer trials (see Figure 1 for the format of presentation). Four out of the five categories, “word *malicious* (present/absent),” “fileserver

(attacked/not attacked),” “operation (successful/unsuccessful),” and “user (inside/outside),” overlapped between the two conditions. Two categories, “alerts (present/absent)” and “files (manipulated/not-manipulated),” were uncommon across conditions. For the overlapping categories, the proportions of usage were somewhat different for the two conditions. However, a significant difference was found only for the “word *malicious*” category, its proportions were greater in the low condition compared with the high condition. As the “word *malicious*” category did not always classify threats correctly in this experiment, the difference in attention between the two conditions might partly explain participants' superior performance in the high condition.

As seen in Table 1, the two leading categories of rules that participants used in the low condition were user location and the presence of the word *malicious*. Similarly, in the high condition, the two categories that participants attended to most were user location and the presence of an alert. The user location and alerts represent categories that could correctly classify an event in this task. In the high condition, participants paid attention to both of these categories among the top five; in the low condition, however, they only paid attention to the user category and ignored the alerts category. This difference in attention likely accounts for the superior performance in the high condition.

A Linear Model: Validating Attention to Situational Attributes

The literature on heuristics and biases shows that linear models have been highly successful in explaining human behavior and leads to approximate correct responses that are more accurate than even expert judgments (Dawes, 1979; Goldberg, 1970). For example, when Dawes and Corrigan (1974) applied different linear models to five different datasets to predict a criterion, an equal weighting linear model (the simplest assumption of linearity) outperformed all other competing models. If the categories and weights (i.e., the relative proportion) reported in Table 1 are representative of our behavioral findings, then they should produce a close fit to the hit and false-alarm rates observed in human data when these categories are simulated in a linear model. To explore this idea further, we developed a stochastic linear model consisting of derived categories and attention weights.

Stochastic Linear Model

We represent the binary decision process of classifying events as threats and non-threats in a trial according to the following rule:

If the model's Outcome $>$ *threshold*, then classify the event as threat; otherwise, classify the event as a non-threat. (1)

The *threshold* is a free parameter that is calibrated in the model. The Outcome is defined according to a linear model:

For the low condition: Outcome = 0.23 * Word *malicious* + 0.15 * Fileserver + 0.11 * Operation + 0.34 * User + 0.17 * Files

Table 1: The top five attention categories and their descriptions in the low and high conditions.

Categories	Low Condition	High Condition	Statistic
	Relative Proportion (%) ¹	Relative Proportion (%) ¹	Low versus High Condition
<i>Word malicious (present/absent)</i> If the description of the event contains the word malicious, then treat the event as a threat; otherwise treat it as a non-threat.	22.51	7.69	$t(36) = -2.34$, $p < .05$, $r = 0.36$
<i>Fileserver (attacked/not attacked)</i> If the description of the event contains an attack on fileserver, then treat the event as a threat; otherwise treat it as a non-threat.	15.37	8.13	$t(36) = -1.45$, ns , $r = 0.23$
<i>Operation (successful/unsuccessful)</i> If the description of the event contains execution of an operation successfully, then treat the event as a threat; otherwise treat it as a non-threat.	10.52	11.13	$t(36) = -0.09$, ns , $r = 0.01$
<i>User (inside/outside)</i> If the description of the event contains a user generating the event from outside the company, then treat the event as a threat; otherwise treat it as a non-threat.	34.01	53.43	$t(36) = -1.57$, ns , $r = 0.25$
<i>Alerts (present/absent)</i> If the description of the event is accompanied by an alert, then treat the event as a threat; otherwise treat it as a non-threat.	-	19.59	-
<i>Files (manipulated/not-manipulated)</i> If the description of the event details files being manipulated, then treat the event as a threat; otherwise treat it as a non-threat.	17.56	-	-
Total	100%	100%	

Note: The r is the effect size.

For the high condition: Outcome = 0.08 * Word *malicious* + 0.08 * Fileserver + 0.11 * Operation + 0.53 * User + 0.20 * Alerts (2)

Where the “Word *malicious*,” “Fileserver,” “Operation,” “User,” “Files,” and “Alerts” are dummy variables (taking a values of 0 or 1) corresponding to the five categories in Table 1. The weights (i.e., coefficients) that multiply the dummy variables are the relative proportions of the respective categories reported in Table 1. The model is “stochastic” because the exact value of a dummy variable (0 or 1) for each event in a trial depends upon comparing $U(0, 1)$ with the dummy variable’s *attention probability* parameter. The rule for paying attention to a dummy variable in the model is the following:

If the category is applicable (i.e., present in the event's a description) to the network event and $U(0, 1) \leq$ dummy variable’s *attention probability*, then the dummy variable equals 1; otherwise, the dummy variable equals 0 (3)

Each dummy variable’s *attention probability* is a free parameter that is calibrated in the model. This parameter represents whether a model participant pays attention to a

category when it is present in the event’s description or in the accompanying alert. Furthermore, if the category is attended to, then the Outcome takes a weighted contribution of the category into the binary decision. Also, more than one category could be attended to (dummy variable = 1) for an

encountered event. Therefore, the model captures the property that human participants might stochastically pay attention to multiple categories for an encountered event.

Parameter Calibration

The model’s free parameters, i.e., each dummy variable’s *attention probability* and *threshold*, were calibrated to human data in the low and high conditions, separately. Calibrating the model to human data means running it in the same training conditions experienced by human participants to find the parameters values which minimize the sum of mean squared deviations (Sum of MSDs) between the model’s hit and false-alarm rates and human hit and false-alarm rates, respectively. The smaller the sum of MSDs, the better the model’s ability to capture human behavior is. The model

was calibrated separately to training trials in the low and high conditions using a genetic algorithm program. To calibrate the model, we varied the *threshold* and *attention probability* parameters between 0.0 and 1.0 (their minimum and maximum values). The model was run using the same number of simulated participants as the number of human participants that participated in the two conditions.

Table 2 presents a summary of the calibrated parameters and the models’ performance (MSD) in the two conditions. The model performed reasonably well to capture the hit and false-alarm rates in both conditions; however, it was slightly better in capturing the false-alarm rate than the hit rate. The calibrated value of the *threshold* parameter was found to be similar in both conditions (close to 30%). Moreover, based upon the *attention probability* parameters, the model seemed to frequently attend to the correct categories, “user” and “alerts,” in the high condition to make decisions. In fact, the model’s attention to the user category was greater in the high condition (.76) than in the low condition (.50). Figure 3 shows the model fits to human data in both conditions for training and transfer.

The MSDs between the model and human hit rates at transfer in the low and high conditions were 0.0003 and 0.0002, respectively. Similarly, the MSDs between the model and human false-alarm rates at transfer in the low and high conditions were 0.0001 and 0.0052, respectively. These MSDs are very small and therefore the model provides a good approximation to the human transfer performance.

Table 2: Summary of calibration of the linear model during training.

Condition	Parameters	MSD (Hit Rate)	MSD (False-Alarm Rate)
High	<i>threshold</i> = .31; <i>attention probability</i> (Word <i>malicious</i>) = .10; <i>attention probability</i> (Fileserver) = .99; <i>attention probability</i> (Operation) = .44; <i>attention probability</i> (User) = .76; <i>attention probability</i> (Alerts) = .53	0.0112	0.0016
Low	<i>threshold</i> = .33; <i>attention probability</i> (Word <i>malicious</i>) = .53; <i>attention probability</i> (Fileserver) = .60; <i>attention probability</i> (Operation) = .83; <i>attention probability</i> (User) = .50; <i>attention probability</i> (Files) = .30	0.0130	0.0071

Predictions in Novel Transfer Conditions

Typically, one could think of cyber-threats as rare events in the real world (Jajodia et al., 2010). If these threats occur rarely at transfer, then participants trained in the low condition might benefit more from their training. That is because, the experiences gained in the low base-rate training condition are likely to be more suited to the rare transfer condition compared with those gained in the high condition. One way to test this expectation is by creating a rarer transfer trial (i.e., whose base-rate is less than that in the low condition’s training trials and that in the original transfer trial). One such rare transfer trial could have a threat base-rate of 4% (i.e., only 1 event out of 25 events is an actual

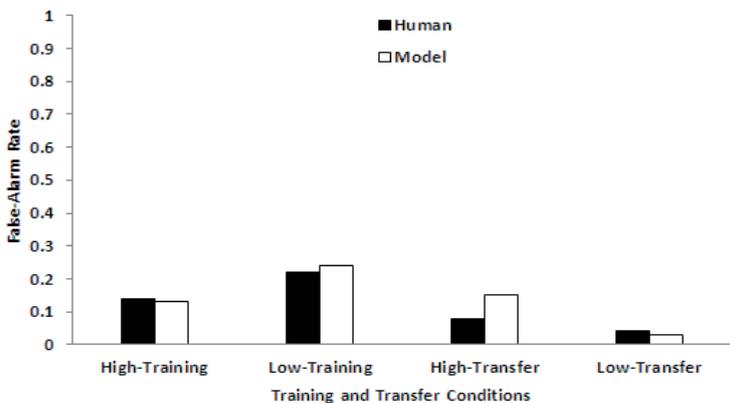
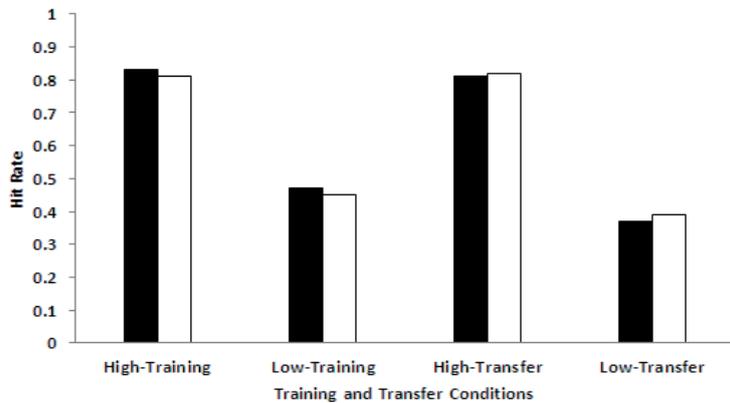


Figure 3: The model and human hit rate and false-alarm rate in the low and high conditions during training and transfer.

threat in the trial).

In the transfer trial of our experiment, the human hit rate in the high condition was 81% while it was 37% in the low condition (i.e., a gap of 44%; see Figure 2). In the transfer trial, the human false-alarm rate in the high condition was 8% while it was 4% in the low condition (i.e., a gap of 4%). Thus, we expect the accuracy to be greater in the high condition than in the low condition; however, based upon the discussion above, it is also possible that people trained in the low condition will perform better in the rare transfer trial (with a 4% threat base-

rate). Therefore, we expect the model’s hit rate predictions in the low condition to increase and in the high condition to decrease at transfer, closing the overall gap. However, for the same rare transfer trial, we expect the model’s false-alarm rate predictions in the low condition to decrease and in the high condition to increase at transfer, widening the overall gap.

Predictions generated from our calibrated model were in agreement with these expectations. The model’s performance in the rare transfer trial showed a hit rate in the high and low conditions to be 76% and 46%, respectively. Therefore, its overall performance in terms of hit rate was superior in the high condition compared with the low condition; however, the gap between the hit rates in the two conditions was reduced to 29%. Similarly, the model’s performance showed a false-alarm rate in the high and low conditions of the rare transfer trial to be 16% and 4%, respectively (i.e., an increased gap of 11% as expected). Overall, the gap between hit and false-alarm rates in the low and high conditions moved in the direction as expected above.

Another aspect to consider is the length of analysts’ training sessions. In the real world, lengthy training might become costly because of resources and time (Kanellis, 2006). In such situations, one method to save costs is to reduce the training length and evaluate cyber-threat detection accuracy at transfer. Here, we derive predictions

Table 3: Summary of half calibration length of the linear model during training.

Condition	Parameters	MSD (Hit Rate)	MSD (False-Alarm Rate)
High	<i>threshold</i> = .33; <i>attention probability</i> (Word <i>malicious</i>) = .20; <i>attention probability</i> (Fileserver) = .71; <i>attention probability</i> (Operation) = .93; <i>attention probability</i> (User) = .96; <i>attention probability</i> (Alerts) = .53	0.0149	0.0004
Low	<i>threshold</i> = .29; <i>attention probability</i> (Word <i>malicious</i>) = .44; <i>attention probability</i> (Fileserver) = .71; <i>attention probability</i> (Operation) = .94; <i>attention probability</i> (User) = .46; <i>attention probability</i> (Files) = .88	0.0208	0.0073

from the linear model in a situation where the training in both the low and high conditions is reduced by halving its original length (i.e., only first 5 trials long), and the model is then transferred to the rare transfer trial with a 4% base-rate. Table 3 provides the summary of the calibrated parameters and the MSDs to the halved training length.

Again in the high condition, the free parameters have greater values in the “user” and “alert” categories compared to those in the low condition. Furthermore, the MSDs in Table 3 are slightly higher compared with those reported for full-length training in Table 2. At transfer, the calibrated model’s hit and false-alarm rates were 67% and 29% in the high condition, respectively; whereas, these rates were 37% and 14% in the low condition, respectively. When these proportions are compared with those reported above for the rare transfer trial with full-length training, we find a drop in hit rate as well as an increase in false-alarm rate in both conditions. Thus, our model predictions suggest that reducing training conditions by half their original length is likely to save time and costs, but also likely to decrease the analysts’ detection accuracy at transfer.

Discussion

In this paper, we evaluated the effects of training security analysts in conditions with different threat base-rates (low and high) and transferring them to novel conditions (that were either in-between those encountered during training or possessing a very low base-rate). We found that their transfer performance is superior when their training environments provide them with lengthy training and higher threat base-rates. That is likely because higher base-rates allow participants to form improved hypotheses about threats that they could test during their training and transfer performance (Dutt & Gonzalez, in press). This reasoning is clearly reflected in the greater proportions of calibrated attention probability for the “user” and “alert” attributes (i.e., the attributes that reveal the ground truth) in the high condition compared with in the low condition.

Our results suggest that any training interventions for analysts should pay close attention to how the base-rate of threats compare to their actual work conditions. Also, the length of training (e.g., weeklong or half a week) is likely to influence analysts’ learning and performance at transfer. Thus, the training length is likely to affect their performance in actual work conditions. Generally, it would be advisable to keep the training extended in length, as well as train analysts on scenarios that makes them experience a high threat base-rate. In fact, the linear model could be used to derive the optimal length of training session for a desired level of accuracy. Although we can only speculate, but our results are also likely to be valid for other emergency situations like training miners for a low-probability mine emergency, or training air-traffic controllers for low probability air accidents.

In this paper, we contribute to the growing literature on cyber security by evaluating the benefits and costs of

training analysts in scenarios that differ in threat base-rates. Although base-rates were different, other aspects of the scenario (i.e., the sequence of attack, computers compromised, etc.) were identical. Thus, future research is likely to benefit from our results by manipulating other aspects of attack scenarios and evaluating the influence on training and transfer interventions. Also, as the linear model might be more mathematical than cognitive in its formulation, future research is also likely to benefit by comparing how other cognitive models, which use memory and activation (including some data-mining algorithms), perform compared to the linear model. Finally, we also contribute a method of going from an experiment about detecting cyber threats to developing a cognitive model based upon participants’ self-reported strategies. This “model discovery” approach that uses human data to construct cognitive models might provide useful insights for alternative modeling approaches to model development.

Acknowledgements

This research was a part of a MURI Award on Cyber Situation Awareness (MURI; #W911NF-09-1-0525) from ARO to Cleotilde Gonzalez.

References

- Obama, B (2011, May). Remarks by the President on securing our nation's cyber infrastructure. Retrieved from http://www.whitehouse.gov/the_press_office/Remarks-by-the-President-on-Securing-Our-Nations-Cyber-Infrastructure/
- D’Amico, A., Whitley, K., Tesone, D., O’Brien B., & Roth, E. (2005). Achieving cyber defense situational awareness: A cognitive task analysis of information assurance analysts. *Proceedings of the Human Factors And Ergonomics Society Annual Meeting*, 49(3), 229-233.
- Dawes, R. M. (1979). The robust beauty of improper linear models. *American Psychologist*, 34, 571-582.
- Dawes, R.M., & Corrigan, B. (1974). Linear models in decision making. *Psychological Bulletin*, 81, 95-106.
- Dutt, V., Ahn, Y. S., & Gonzalez, C. (2011). Cyber situation awareness: Modeling the security analyst in a cyber-attack scenario through Instance-Based Learning. *Lecture Notes in Computer Science*, 6818, 280-292. doi: 10.1007/978-3-642-22348-8_24.
- Dutt, V., & Gonzalez, C. (in press). Cyber situation awareness: Modeling the security analyst in a cyber attack scenario through Instance-based Learning. In C. Onwubiko & T. Owens (Eds.), *Situational Awareness in Computer Network Defense: Principles, Methods and Applications*. Hershey, PA: IGI Global.
- Goldberg, L. R. (1970). Man vs. model of man: A rationale, plus some evidence for a method of improving clinical inferences. *Psychological Bulletin*, 73(6), 422-432.
- Gonzalez, C., & Dutt, V. (2011). Instance-based learning: Integrating decisions from experience in sampling and repeated choice paradigms. *Psychological Review*, 118(4), 523-551.
- Hertwig, R., & Erev, I. (2009). The description-experience gap in risky choice. *Trends in Cognitive Sciences*, 13(12), 517-523.
- Jajodia, S., Liu, P., Swarup, V., & Wang, C. (2010). *Cyber situational awareness*. New York: Springer.
- Kanellis, P. (2006). *Digital crime and forensic science in cyberspace*. New York: Idea Group Publishing.
- Sideman, A. (2011, February 23). Agencies must determine computer security teams in face of potential federal shutdown. *Federal Computer Week*. Retrieved from <http://fcw.com/articles/2011/02/23/agencies-must-determine-computer-security-teams-in-face-of-shutdown.aspx>
- Zakaria, T. (2011, October 1). Government simulates cyber attack for training. *Reuters*. Retrieved from <http://www.reuters.com/article/2011/10/01/us-usa-cyber-idaho-idUSTRE78T08B20111001>

Exploring Feature Collocation for Semantic Concept Identification

Meng Yu (myu3@uci.edu)

Department of Mathematics,
University of California, Irvine, CA, 92697, USA

Keywords: feature collocation, semantic concept identification, l_1 regularization.

Problem Description

Motivated by object representation in psychology, we present a binary feature classifier for the purpose of semantic concept category identification/classification by incorporating feature distribution. We propose the classification algorithm based on the variant of l_1 norm regularized sparse classifiers, where the features are weighted according to their distribution, which is estimated by “maximum collocation”. This method achieves high accuracy in identifying semantic concepts, outperforming standard benchmark methods on a large database of animal and artifact features.

Suppose your friend tells you they are thinking of a particular mammal animal, asks you what type it is, and starts listing its features: it has a tail, has four legs, can’t swim, and so on. You are now faced with a *category identification* problem, which requires you to infer the most likely category of an instance given knowledge of some of its features (Kemp, Chang, & Lombardi, 2010). Category identification offers an interesting window onto the structure of mental representations, since it involves the relationship between categories and features, and so requires the representation of both what makes instances different, and what makes them the same. One of the main shortcomings of existing classification work is that feature importance has not been well investigated (Zhang, Yu, Lee, & Xin, 2011). Features are often preselected from the beginning which actually do not equally or positively contribute to the performance of classification. However, not all of the features will be important to an object’s representation. Thus, weighting features without adversely affecting the performance is an important task for classification.

Feature Distribution and Weighting

We propose to weight the features such that categories can be differentiated more efficiently according to the binary feature’s distribution¹. This is motivated by stimuli representation in psychology (Jones, 1983) since it was studied that people identify the semantic concepts by choosing features in a systematic way. One task is to choose important features by how useful they are in distinguishing categories. For example, in mammal domain, feature “is pregnant” is less important than “has long neck”. This empirical motivation becomes the principle for feature importance measure.

Maximum collocation is described here for measuring the feature importance based on two heuristics (Zeigenfuse &

Table 1: Representative features illustrating behavior of the usefulness measures. Black dot means that the instance has the corresponding feature.

	C 1	C 2	C 3	Cue	Cat.	Colloc.
f 1	• • •			1	1	1
f 2	• • •	•		3/4	1	3/4
f 3	• •			1	2/3	2/3
f 4	•			1	1/3	1/3
f 5	• • •	• •	• •	3/7	1	3/7
f 6	• • •		• • •	1/2	1	1/2

Lee, 2010). The first of these is maximum cue validity, defined as the maximum over categories c_j ($j = 1, 2, \dots, n_c$) of cue validity, the probability an instance belongs to c_j given that it has a feature f , $p(c_j|f)$. We also look at maximum category validity, defined similarly as the maximum over categories c_j of the category validity, the probability an instance has a feature f given that it belongs to c_j , $p(f|c_j)$. Finally, the maximum collocation is the maximum over categories c_j of the collocation, the product of a feature’s cue and category validities, $p(c_j|f)p(f|c_j)$. Maximum collocation is a measure of how simultaneous concentrated in and diffuse across a category a feature is. Features with high maximum collocation are associated with most instances within a category and few outside it, as illustrated by Feature 1 in Table 1. Alternatively, Features 4 and 6 show why it is necessary for both of these to be true. Those features associated with only a small fraction of instances within a single category will have high maximum cue validity but low maximum category validity (Feature 4). Those features possessed by most instances in more than one category will have high maximum category validity but low maximum cue validity (Feature 6).

Collocation Weighted Classifiers

The motivation for using sparse representation (SR) for category identification is that SR adaptively selects the relevant support data points from the training data, allowing us to identify the semantic concept using a few relevant examples from the training dataset, and alleviating adverse effects of instances variability in the training dataset. Mathematically, in a typical SR formulation, a *dictionary* D is constructed as $D = [d_1, d_2, \dots, d_n]$, where each $d_i \in R^m$ is a feature vector of i th instance. To represent a test instance in terms of its feature vector y , SR solves the equation $y = D\theta$, where a regularization is enforced on θ , such that only a small number

¹Feature value is “yes” or “no”

² n_c is the number of categories.

of instances from the dictionary D are selected to describe y . Sparsity regularization helps the representation to rule out irrelevant instances and be insensitive to within-category variability in the dictionary. The test instance is assigned to the category with the smallest residual in representing y as a linear combination using all instances from that category.

With the feature importance measure, features are weighted by maximum collocation. Denote U_{col} as the diagonal matrix with $u_{col}(f)$ as the diagonal entries, where $u_{col}(f)$ is the maximum collocation for feature f . The weighted dictionary and test instance become $U_{col}D$ and $U_{col}y$ respectively. We developed three SR variants for the classification. Then, as a result, all the features contribute unequally in the sparse representation. The conventional SR optimization is given by

$$\theta^* = \arg \min_{\theta} \frac{1}{2} \|U_{col}y - U_{col}D\theta\|_2^2 + \mu \|\theta\|_1, \quad (1)$$

where μ is the trade-off parameter.

Due to the non-negativity of the features, the above l_1 regularized unconstrained convex optimization (1) becomes a non-negative penalized l_1 regularized constrained convex optimization (2) as below. The non-negative weights in θ indicate the importance of an instance with the natural interpretation that this constraint forces representations that include on instances that provide evidence for a category identification decision.

$$\theta^* = \arg \min_{\theta} \frac{1}{2} \|U_{col}y - U_{col}D\theta\|_2^2 + \mu \|\theta\|_1 \quad s.t. \quad \theta \geq 0$$

Eventually, since we expect sparse representation errors, for which l_1 norm regularization seems to be more appropriate. The optimization (1) is re-formulated as

$$\theta^* = \arg \min_{\theta} \|e\|_1 + \mu \|\theta\|_1 \quad s.t. \quad U_{col}y = U_{col}D\theta + e. \quad (2)$$

Dataset and Evaluations

Our data come from the Leuven Natural Concept Database (DeDeyne, et al, 2008), involving 295 words (i.e. categories), distributed over 11 semantic domains: five animal domains (30 mammals categories, 30 birds categories, 23 fish categories, 26 insects categories, and 20 amphibians&reptiles categories) with 764 animal features, and six artifact domains (31 kitchen utensils categories, 30 clothing categories, 27 musical instruments categories, 29 vehicles categories, 19 weapons categories and 30 tools categories) with 1295 artifact features. Features used to describe those words include perceptual, functional characteristics and any other background information that applies. Most importantly for our modeling, the words (i.e. semantic concept categories) and features were combined in a feature verification task, in which four participants judged whether or not each of the features belonged to each of the words. In the experimental evaluations, we split the data into training and test sets for a 4-fold

cross-validation. In each validation, we train the classifier using data from three participants and test on the participant that is left out, i.e. a leave-one-out cross-validation. We use the set of features a participant assigned to a word — “can fly”, “is small”, and so on — as the input to a category identification problem, for which the task is to identify the category associated with that list of features.

The identification accuracy for the proposed variants of weighted sparse representation methods achieves 84 percent in average, outperforming typical classification methods, such as k nearest neighbor, logistic regression and decision tree. To examine the performance variation of the proposed feature collocation based classifiers on different categories and domains, we compute an overall rank of average residual for each category, shown in Fig. 1 as an example. Large magnitudes suggest ambiguities in semantic category identification. The identification errors for mammals is very small, but large for amphibians&reptiles domain. We believe the proposed approach constitutes a useful starting point for understanding how people do semantic concept category induction.

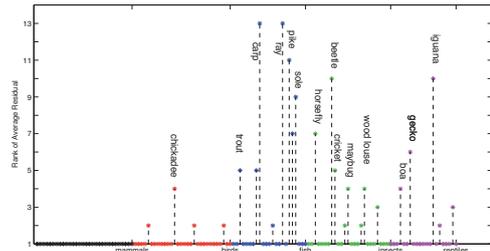


Figure 1: Performance variation for semantic concepts in animal domains.

References

- DeDeyne, S., Verheyen, S., Ameel, E., Vanpaemel, M., Dry, M., & Voorspoels, W. (2008). Exemplar by feature applicability matrices and other dutch normative data for semantic concepts. *Behavior Research Methods*, 40, 213–231.
- Jones, G. V. (1983). Identifying basic categories. *Psychological Bulletin*, 94, 423–428.
- Kemp, C., Chang, K. M., & Lombardi, L. (2010). Category and feature identification. *Acta Psychologica*, 133, 216–233.
- Zeigenfuss, M. D., & Lee, M. D. (2010). Heuristics for choosing features to represent stimuli. In *Proceedings of the 32nd annual conference of the cognitive science society* (pp. 1565–1570).
- Zhang, S., Yu, M., Lee, M. D., & Xin, J. (2011). Modeling category identification using sparse instance representation. In *Proceedings of the 33rd annual conference of the cognitive science society*. Austin, TX: Cognitive Science Society.

Uses, Abuses and Misuses of Computational Models in Classical Conditioning

Eduardo Alonso (E.Alonso@city.ac.uk)
School of Informatics, City University London
London EC1 0HB, UK

Esther Mondragón (e.mondragon@cal-r.org)
Centre for Computational and Animal Learning Research
St Albans AL1 1RQ, United Kingdom

Abstract

Classical conditioning is at the heart of most learning processes. It is thus essential that we develop accurate models of conditioning phenomena and data. In this paper we review the different uses of computational models in exploring conditioning, as simulators and as psychological models by proxy.

Keywords: Classical conditioning; computational models; psychological models; simulation.

Introduction

It is universally accepted that conditioning is at the basis of most learning phenomena: indeed, models of classical and instrumental conditioning have proved to be relevant to human and non-human learning both theoretically and in practice (Wasserman & Miller, 1997; Pearce & Bouton, 2001; Hall, 2002; Schachtman & Reilly, 2011). In this enterprise, collaboration between computer scientists and psychologists has enjoyed considerable success (Schmajuk, 2010a; Alonso & Mondragón, 2011): connectionist models have been used to better predict discrimination and categorization phenomena (Shanks 1995); in addition, it has been argued that classical conditioning rules can be naturally interpreted as an instance of more comprehensive computational neuroscience models (Dayan & Abbott, 2001; Schmajuk, 2010b).

This collaboration is sustained on various arguments: expressing models in the form of algorithms provides us with formal ways of representing psychological insights and of calculating their predictions accurately and quickly; from computational models we also borrow a view on how information is processed, a computer analogy that has proved useful in understanding cognition; moreover, the underlying architectures of computational models, for instance the hidden units of an artificial neural network or the way feedback is computed in recurrent networks, resemble the mechanics of associative learning at both the neural and conceptual levels; finally, machine learning models, such as temporal difference learning and Bayesian learning, can be understood as effective abstractions of the way associations are formed and computed.

In this paper we analyse critically the assumptions upon which such arguments are built. We identify two main trends in so-called computational psychology, more in particular in the use of computational models in the study of conditioning, namely, as simulators and as

psychological models in themselves, and evaluate their respective merits.

Computational Models as Simulators

Firstly, a computational model can be understood to be an implementation of a (pre-existing) psychological model. Simulations serve two main purposes: On the one hand, implementing a model requires precise definitions –be it in the form of a specific programming language or as a formal model, that in turn makes the original psychological model “accountable”. On the other hand, algorithms allow us to execute calculations rapidly and, most importantly, accurately. Automation is critical, particularly when the models are described in non-linear equations that can only be solved numerically as it is the case of recent psychological models of conditioning (Balkenius & Morén, 1998; Vogel *et al.*, 2004; Mitchell & Le Pelley, 2010; Alonso & Schmajuk, 2012). Crucially, the outputs of a simulation feedback the psychological models –thus becoming an essential part of the cycle of theory formation and refinement.

It is worth noting though that the benefits derived from using implementations do not spring exclusively from the formal specification of the psychological models in equations and algorithms. Per se, such descriptions constitute a mathematical model, a necessary yet no sufficient condition for a formal model to be computational. The essence of a computational model lies in the fact that it is implemented. According to this view, in psychology, the same as in computational physics and in computational biology, a computational model is a model that has been simulated.

This view is not without detractors: It has been argued that a model is computational if it is “implementable” – even if it was not originally described as a full-bodied computational model. We think that this is an *abuse* of the term computational since any psychological model of conditioning would fit this definition. To use a parallelism: this use of the term “computational” would make all models in Physics since Galileo’s computational.

This brings up a subtler issue: We are using the term computational model in a “modern” sense. Indeed, a computational model is just a formal model of computation and “computation” does not necessarily require its implementation in a computer. Mathematically, the notion of computation is a formalization of the concept of algorithm, a mechanical or automated procedure to prove theorems proposed by Alan Turing to attack Hilbert’s *Entscheidungsproblem* (Turing, 1937). Modern

computers are mere physical instantiations of the abstract machines that would compute such procedures. But they don't play a fundamental part in the definition of computation. Indeed, such definition was proposed well before the first digital general-purpose computers had even been designed. Contrarily, our position is that a computational model needs to be implemented in a computer. Otherwise, a computational model does not add anything to what constitutes a mathematical model in its own right.

We would also like to comment on a second potential source of confusion about the term computational –that comes from cognitive science rather than from mathematics. The term “computational” has been linked to David Marr’s Tri-Level Hypothesis on vision where the “what” refers to the computational level, the “how” to the algorithmic level and the “where” to the implementational level (Marr, 1982). However insightful such analysis may be, clearly what Marr referred to as “computational” is “psychological” –when applied to cognition. Insisting on talking about psychological models as if they were computational based on such taxonomy is, in our opinion, a source of misunderstanding.

Computational Models as Psychological Models

The second use of the term “computational” is more controversial: it is claimed that a computational model can be considered a psychological model in itself. We argue that this position, a milestone in cognitive science and artificial intelligence, is a *misuse* of the term. Let’s illustrate our contention using a paradigmatic example: The use of Artificial Neural Networks (ANNs) in the study of conditioning has been advocated at several, inter-related levels that we are now analyzing.

Ontological Level

ANNs are considered material models of conditioning. The underlying reasoning is that (a) ANNs model by analogy natural neural networks and that (b) psychological processes, including conditioning, are ultimately embedded in natural neural networks; hence, indirectly, ANNs model conditioning.

Notwithstanding the popularity of this line of argumentation, it is widely acknowledged that ANNs do not resemble natural neural networks in any fundamental way (Enquist & Ghirlanda, 2005); besides, there is no strong evidence suggesting that electrical or chemical neural activity and conditioning are related (Morris, 1994) –or for that matter, that psychological processes can be localized in specific brain regions as recently exposed in (Vul *et al.*, 2009), but already advanced in (Uttal, 2001).

Even if it did, a neural analysis would not necessarily be the right level to study learning phenomena. In the words of Burrhus F. Skinner “The analysis of behavior need not wait until brain scientists have done their part. The behavioral facts will not be changed (...). Brain scientists may discover other kinds of variables affecting behavior, but they will turn to a behavioral analysis for the clearest account of the effects of these variables” (Skinner, 1989, pp. 18). Regardless of the antipathy that Skinner’s radical

behaviorism provokes among neuroscientists such a statement does not contradict a version of reductionism that most of them would endorse, namely, Richard Dawkin’s hierarchical reductionism (Dawkins, 1986).

Formal Level

Relatedly, that a version of Dirac’s rule can be taken as a model of both neural plasticity and long-term potentiation effects –the Hebbian rule (Hebb, 1949)– and association formation –for example, Rescorla and Wagner’s rule (Rescorla & Wagner, 1972)– cannot be considered as proof of any common underlying structure and should not be used as an argument to reduce psychological phenomena to their alleged neural substratum.

Likewise, that Rescorla and Wagner’s rule is essentially identical to the Widrow-Hoff rule (Widrow & Hoff, 1960) for training *Adeline* units and that, in turn, such a rule can be seen as a primitive form of the generalized delta rule for backpropagation only tells us that, computationally speaking, associative learning follows an error-correction algorithm¹. What a computational model does not tell us, however, is which underlying psychological processes (attention, motivation, etc.) intervene in conditioning or how the physical characteristics of the units involved (e.g., the salience of the stimuli) affect such processes.

Clearly, sharing a common formal expression does not imply that the phenomena so expressed are of the same nature: for instance, power functions can be used to express the relationship between (1) the magnitude of a stimulus and its perceived intensity (Stevens’ law), (2) the metabolic rate of a species and their body mass (Kleiber’s law), and (3) the orbital period of a planet and its orbital semi-major axis (Kepler’s third law). Stressing this point, allow us to quote Richard Shull: “The fact that an equation of a particular form describes a set of data does not mean that the assumptions that gave rise to the equation are supported. The same equation can be derived from very different sets of assumptions” (Shull, 1991, pp. 246).

Put it another way, if the meaning of a mathematical/formal model is in the linguistic expression it takes (that is, if there is a unique isomorphism between phenomena and algorithms) then either (a) we cannot explain how a theory can be expressed in different sets of equations or (b) we will not be sure about the effect the addition or the removal of a simple parameter may have. Paraphrasing (Chakravartty, 2001), theories and models can be given linguistic formulations but theories and models should not be identified with such formulations.

Representational Level

ANNs are connectionist models according to which information is not stored explicitly in symbols and rules but rather in the weights (strengths) of the connections; learning would consist of changes in such weights. It is claimed, rightly, that these are precisely the assumptions

¹ Incidentally, backpropagation is merely a mathematical procedure to deriving partial derivatives –that was originally proposed to model nationalism and social communications not neural networks (Werbos, 1974).

on which models of conditioning are based and hence, wrongly, that ANNs are an ideal candidate to model conditioning phenomena. This quite straightforward argument is, in fact, a fallacy: As connectionists (at least implementational connectionists) themselves concede the way we represent learning, either as continuous changes of weighted connections or as the result of discrete symbolic processing, is *a matter of convenience* and therefore irrelevant to the study of the structures involved.

Interestingly, this debate has centered in the difference between associative models and computational models of conditioning (Leslie, 2001): It is understood that associative models are historically and conceptually linked to connectionism (Medler, 1998) whereas computational (aka symbolic) approaches take their ideas from information processing (Gallistel & Gibbon, 2001). We don't think that such technical distinction is fruitful and rather agree with Peter R. Killeen in identifying both approaches as formal (Killeen, 2001): Turing machines and ANNs (as well as RMA machines, the Game of Life, and any programming language) are both computational models²; in particular, Turing machines and ANNs are equivalent in their input/output behaviour, that is, they compute the same problems and accept the same languages (in terms of the Chomsky hierarchy (Chomsky, 1956))³.

Functional Level

ANNs typically approximate solutions by iteratively minimizing an error function. And this can be understood as a type of learning that resembles learning by "trial and error" of which associative learning is an example. However, it is worth emphasizing that ANNs merely implement numerical *methods*. Under a misleading name, they are just statistical tools –and, for that matter, certainly not the simplest, fastest or most efficient ones (see, e.g., Mitchie *et al.*, 1994). On the other hand, conditioning models such as Rescorla and Wagner's express dynamic *laws*: Against public opinion, animals do not make predictions and iteratively update an associative value through error minimization towards an optimal one. The associative value at a given time is the right associative value –that exactly describes to which extend the CS has become associated to the US. Let's put it another way: in standard conditions, if the animal "learned" a CS-US association after one single exposure then the animal would be wrong and its corresponding behaviour unadaptive (unless, of course, we exposed it to a very salient US like in flavour-aversion learning). That the system described by Rescorla and Wagner's rule is limited by an asymptote (the reinforcing value of the US) does not confer any special status to such value –rather it defines a constraint of the system.

² It should be noticed moreover that the first mathematical models of (A)NNs, in particular McCulloch and Pitts's (McCulloch & Pitts, 1943) and Turing's B-type machines (Turing, 1948) were intended to formalize logically, i.e., symbolically, the notion of learning.

³ Provided that the values of the weights are restricted to rational numbers (Orponen, 1994).

Structural Level

We are told that the layout of an ANN, the way units are connected between layers, can be seen as a cognitive architecture and, as such, as a psychological model. Let's take a computational example to counter-argue this point: in computer science network communication is modeled according to the so-called Open Systems Interconnection model (OSI) (Zimmerman, 1980), moving from the physical layer that describes the electrical specifications of the devices the networks consist of up to the application layer that describes how the user interacts with a given piece of software. The question is: Why don't we use the OSI model as a psychological model? At the end of the day, structurally, OSI would make as good a psychological model as an ANN. In fact, the OSI model implements a hierarchical and integrated architecture, that is, the type of cognitive architecture that a computational model should allegedly support (Sun, 2008). Thus that ANNs are networks implemented in architectures that take advantage of massive computational parallelism – not surprisingly, the *new connectionism* landmark paper introduced the Parallel Distributed Processing paradigm in cognition (Rumelhart & McClelland, 1986), does not confer them any psychological advantage: Any complex network would do (Newman *et al.*, 2006).

Philosophical Issues

A final more general reason to explain the appeal of computational models in psychology rests on the idea that both computers and the brain are information processing systems, instantiations of a universal Turing machine or any other model of computation. But this alone does not justify the support the "computer metaphor" enjoys. After all, any phenomena can be expressed in terms of some sort of computation. If this analogy is such a powerful metaphor is because it is deeply rooted in Western philosophy and the mechanization of (formal) reasoning, reformulated in the twentieth century in terms of computation. That computation has been effectively embedded in computers has reinforced the idea that so it is in the brain, that the study of the former will help understand the latter and, in a *tour the force*, that computers may be capable of displaying intelligence. Indeed, every scientific theory is shaped in the context of its Age's achievements and prejudices: Like Newton's laws of mechanics strengthened the view of a deterministic Universe that worked as the sophisticated clocks popular at the time our conception of the mind as an information processing machine has certainly been influenced by the development of computing technology.

And precisely because of its generality the information processing model is not necessary or sufficient: working physicists do not model electrons, atoms or galaxies as information processing entities –be it in the form of a cellular automaton as envisaged in (Zuse, 1969) or as a participatory universe (Wheeler, 1990). On the other hand, neither (computational) physicists nor the public would presume that the simulation of a nuclear reaction generates real energy or that a flight simulator really flies. Of course, this does not preclude physicists from theorizing about what type of information is contained in a

physical system (see, for example, literature on quantum entanglement or black holes) or about exploring the physical limits of computers (pioneered by Richard Feynman (Hey & Allen, 2000) and followed up in contemporary theories of quantum computing (e.g., Vedral, 2006)).

Model Selection

The discussion on what a computational model of psychology constitutes affects how we select models and in turn may help us determine what a computational model “truly” is.

Selecting a model, psychological or otherwise, described in natural language or mathematically, is a difficult task that relies in formal definitions and methods as well as on scientific practice and common sense (Kuhn, 1962; Feyerabend, 1975). Indeed, quantitative formulas have been proposed to compare models based on the average size of the deviations from predicted values, the number of data points and the number of free parameters (Akaike, 1974; Schwarz, 1978). However, relying exclusively on such formalisms or applying blindly Occam’s razor is not advisable –evaluating a model requires good judgment based on careful consideration of many factors, both technical and logical (Baum, 1983). The very essence of a model refers to the choices scientists make –choices that reflect what they consider relevant beyond the mere quantitative.

Nonetheless, this analysis begs the question: when we assess computational models of psychology, what do we assess?

If computational models are simulators we would need to select amongst them according to their computational complexity, that is, according to the time and space they take to make computations –complexity that is related to but not reducible to the algorithms they implement. In addition, computing tools must be tested for reliability and dependability against failures –which, in turn, depends on various factors such as programming languages, operating systems, memory capacity, processing speed, as well as on software engineering and management requirements. Computational models as simulators add a new level of sophistication. But this sophistication comes at a price: a computer program is not as “aseptic” as a mathematical description. A computer program comes to life in algorithms and data structures that must comply with software and hardware specifications.

On the other hand, if computational models are considered as a valid alternative to psychological models, which criteria should we use to evaluate them and to choose amongst them? Psychological criteria? There is no clear answer to this question.

Conclusions

To sum it up, although the need to get influx from “outsiders” is recognized within the psychological community (see Townsend, 2008) computational models should be taken with caution. Computational models may provide us with complementary idealized models of psychological phenomena and with powerful statistical tools to construct models of psychological data but they

alone are not the appropriate instruments to answer psychological questions. This is an obvious, hardly original, conclusion –and yet more often than not we read flamboyant news about robots that learn, think and experience emotions and about ANNs that can do anything psychological models can do only better. On the other hand, given the increasing complexity of psychological models developing accurate and rapid simulators to test their predictions is, in our opinion, a must that should take a prominent place in the psychology curriculum.

An extreme case of the use of computational models as psychological models is exemplified in what we call the “engineering” approach: We take psychological data and build a program that fits it. Since the data is psychological, it is argued, the program must constitute a psychological model –confounding subject and method. Another variant of this approach is to propose models of machine learning as psychological models of learning. As an illustration, simple programs that, under very specific conditions, learn mundane tasks by maximizing a reward signal by trial and error have been presented as a “theory of mind” (Sutton, 2003). History has taught us that this kind of hype does not make any good.

To summarize: The adjective “computational” in computational physics or computational biology refers to the use of computational tools, typically simulators and numerical processors but also data mining and data analysis techniques, to study data and phenomena as well as to assess the predictive power of theories and models. We suggest we separate the wheat from the chaff and “limit” the use of the term “computational” the same way when applied to psychology.

References

- Akaike, H. (1974). A new look at the statistical model identification". *IEEE Transactions on Automatic Control*, 19 (6): 716–723.
- Alonso, E. & Mondragón, E. (2011). *Computational Neuroscience for Advancing Artificial Intelligence: Models, Methods and Applications*. Hershey, PA: Medical Information Science Reference.
- Alonso, E. & Schmajuk, N.A. (2012). Special Issue on Computational Models of Conditioning. *Learning & Behavior*.
- Balkenius, C., & Morén, J. (1998). Computational models of classical conditioning: a comparative study. In Mayer, J.-A. , Roitblat, H. L., Wilson, S. W., & Blumberg, B. (Eds.), *From Animals to Animats 5*. Cambridge, MA: MIT Press.
- Baum, W.M. (1983). Matching, Statistics, and Common Sense, *Journal of the Experimental Analysis of Behavior*, 39, 499–501.
- Chakravarty, A. (2001). The semantic or model-theoretic view of theories and scientific realism. *Synthese*, 127, 325-345.
- Chomsky, N. (1956). Three models for the description of language, *IRE Transactions on Information Theory*, 2, 113–124.
- Dawkins, R. (1986). *The Blind Watchmaker*. New York: W. W. Norton & Company.
- Dayan, P., & Abbott, L.F. (2001). *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. Cambridge, MA: MIT Press.
- Enquist, M., & Ghirlanda, S. (2005). *Neural Networks & Animal Behavior*. Princeton, NJ: Princeton University Press.

- Feyerabend, P. (1975). *Against Method*. New York: Verso Books.
- Gallistel, C.R., & Gibbon, J. (2001). Computational Versus Associative Models of Simple Conditioning, *Current Directions in Psychological Science*, 10(4), 146-150.
- Hall, G. (2002). Associative structures in Pavlovian and instrumental conditioning. In H. Pashler, S. Yantis, D. Medin, R. Gallistel and J. Wixted (Eds.), *Stevens' Handbook of Experimental Psychology*, Volume 3, pp 1-45. Hoboken, NJ: John Wiley and Sons.
- Hebb, D.O. (1949). *The Organization of Behavior: A Neuropsychological Theory*. New York: Wiley.
- Hey, T. & Allen, R.W. (Eds.)(2000). *Feynman Lectures on Computation*. Boulder, CO: Westview Press.
- Killeen, P.R. (2001). The Four Causes of Behavior, *Current Directions in Psychological Science*, 10(4), 136-140.
- Kuhn, T. (1962). *The Structure of Scientific Revolutions*. Chicago, IL: University of Chicago Press.
- Marr, D. (1982). *Vision: a Computational Investigation into the Human Representation and Processing of Visual Information*. San Francisco, CA: W.H. Freeman.
- McCulloch, W. and Pitts, W. (1943). A Logical Calculus of Ideas Immanent in Nervous Activity, *Bulletin of Mathematical Biophysics* 5:115-133.
- Medler, D.A. (1998). A Brief History of Connectionism, *Neural Computing Surveys*, 1(2), 18-73.
- Mitchell, C., & Le Pelley, M. (2010). *Attention and Associative Learning: From Brain to Behaviour*. Oxford: OUP.
- Mitchie, D., Spiegelhalter, D.J., & Taylor, C.C., (Eds.) Elder, J.F., IV. (Rev) (1994). Machine Learning, Neural, and Statistical Classification. *Journal of the American Statistical Association*, 91, 436-438.
- Morris, R.G.M. (1994). The neural Basis of Learning with particular Reference to the Role of Synaptic Plasticity: Where Are We a Century after Cajal's Speculations?, in: N.J. Mackintosh, (Ed.), *Animal Learning and Cognition* (pp. 135-183). San Diego, CA: Academic Press.
- Newman, M., Barabási, A-L., & Watts, D.J. (2006). *The Structure and Dynamics of Networks*. Princeton, NJ: Princeton University Press.
- Orponen, P. (1994). Computational Complexity of Neural Networks: A Survey. *Nordic Journal of Computing*, 1(1), 94-110.
- Pearce, J.M., & Bouton, M.E. (2001). Theories of associative learning in animals. *Annual Review of Psychology*, 52, 111-139.
- Rescorla, R.A., & Wagner, A.R. (1972). A theory of Pavlovian conditioning: The effectiveness of reinforcement and non-reinforcement. In A.H. Black & W.F. Prokasy (Eds.), *Classical Conditioning II: Current Research and Theory* (pp. 64-99). New York: Appleton-Century-Crofts.
- Rumelhart, D.E., McClelland, J.L. & the PDP Research Group (Eds.) (1986). *Parallel Distributed Processing*, Vol. 1, Foundations. Cambridge, MA: The MIT Press..
- Schachtman, T.R., & Reilly, S. (2011). *Associative Learning and Conditioning Theory: Human and Non-Human Applications*. Oxford, UK: Oxford University Press.
- Schmajuk, N.A. (2010a). *Computational Models of Conditioning*. Cambridge, UK: Cambridge University Press.
- Schmajuk, N.A. (2010b). *Mechanisms in Classical Conditioning: A Computational Approach*. Cambridge, UK: Cambridge University Press.
- Schwarz, G. (1978). *Estimating the dimension of a model*. The Annals of Statistics, 6, 461-464.
- Shanks, D.R. (1995). *The Psychology of Associative Learning*. Cambridge, England: Cambridge University Press.
- Shull, R.L. (1991). Mathematical description of operant behavior: an introduction. In I.H. Iversen & K.A. Lattal (Eds.), *Experimental Analysis of Behavior* (Vol. 2, pp. 243-282). New York: Elsevier.
- Skinner, B.F. (1989). The Origins of Cognitive Thought. *American Psychologist*, 44(1), 13-18.
- Sun, R. (2008). *The Cambridge Handbook of Computational Psychology*. New York: Cambridge University Press.
- Sutton, R.A. (2003). *Reinforcement Learning's Computational Theory of Mind*, talk delivered at Rutgers, 02/14/03, available at <http://webdocs.cs.ualberta.ca/~sutton/Talks/Talks.html>.
- Townsend, J.T. (2008). Mathematical psychology: Prospects for the 21st century: A guest editorial. *Journal of Mathematical Psychology*, 52, 269-280.
- Turing, A.M. (1937). On Computable Numbers, with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 2 42, 230-65.
- Turing, A.M. (1948). *Intelligent Machinery*, <http://www.turingarchive.org/browse.php/C/11>. It can be found in Copeland (2004), Ince (1992) and Meltzer, B. and Michie, D. (eds.), *Machine Intelligence 5*, Edinburgh: Edinburgh University Press, 1969, 3-23.
- Uttal, W.R. (2001). *The New Phrenology: The Limits of Localizing Cognitive Processes in the Brain*. Cambridge, MA: The MIT Press.
- Vedral, V. (2006). *Introduction to Quantum Information Science*. Oxford, England: Oxford University Press.
- Vogel, E.H., Castro, M.E., & Saavedra, M.A. (2004). Quantitative models of Pavlovian conditioning. *Brain Research Bulletin*, 63, 173-202.
- Vul, E., Harris, C., Winkelman, P. & Pashler, H. (2009). Puzzlingly High Correlations in fMRI Studies of Emotion, Personality, and Social Sciences. *Perspectives on Psychological Science*, 4(3), 274-290.
- Wasserman, E.A., & Miller, R.R. (1997). What's elementary about associative learning? *Annual Review of Psychology*, 48, 573-607.
- Werbos, P.J. (1974). *Beyond regression: New tools for prediction and analysis in the behavioral sciences*. PhD dissertation, Harvard University, Cambridge, MA.
- Wheeler, J.A. (1990). Information, physics, quantum: The search for links, in: W. Zurek (Ed.) *Complexity, Entropy, and the Physics of Information*. Redwood City, CA: Addison-Wesley.
- Widrow, G., & Hoff, M.E. (1960). *Adaptive switching circuits*. Institute of radio engineers, Western Electronic show & convention, Convention record, Part 4, 96-104.
- Zimmerman, H. (1980). OSI Reference Model – The ISO Model of Architecture for Open Systems Interconnection, *IEEE Transactions on Communications*, 28(4), 425-432.
- Zuse, K. (1969). *Rechner Raum*. Braunschweig: Friedrich Vieweg & Sohn.

How WM load influences pronoun interpretation

Jacolien van Rij (J.C.van.Rij@rug.nl)

University of Groningen

Hedderik van Rijn (D.H.van.Rijn@rug.nl)

University of Groningen

Petra Hendriks (P.Hendriks@rug.nl)

University of Groningen

Introduction

We have implemented a computational model to investigate how WM capacity can influence the interpretation of pronouns in a linguistic context. From our cognitive model the prediction follows that adults' comprehension of pronouns can decrease with WM load. We performed an experiment to test this prediction.

How do listeners determine the referent of a pronoun? Generally, pronouns (*he*) are used when reference is intended to the topic, which is the most salient character in the linguistic context. More specific forms such as full noun phrases (*the soccer player*) or proper names (*Eric*) are used when reference is intended to less salient or new characters. Different factors have been found to affect the saliency of characters in the linguistic context, such as the grammatical role. The subject of the previous sentence is likely to be the current topic (Grosz, Weinstein, & Joshi, 1995). As a result, listeners will often interpret a pronoun as referring to the previous subject (a.o., Stevenson, Crawley, & Kleinman, 1994). However, for children up to the age of 7, the grammatical role seems to be a less important cue than for adults (Koster, Hoeks, & Hendriks, 2011). Children's use of grammatical information in pronoun comprehension seems to increase with a higher working memory (WM) capacity score (Koster et al., 2011).

Cognitive model

Within the cognitive architecture ACT-R (Anderson, 2007), we implemented a computational model to simulate the production and comprehension of referring expressions. The model's task is to find the interpretation of a referring expression given the preceding linguistic context.

To find the interpretation of a pronoun, the model needs to know which referent is the current discourse topic. The discourse topic is modeled as the referent with the highest saliency, i.e., with the highest activation in declarative memory. The activation of referents is dependent on the preceding discourse (i.e., frequency and recency), but is also influenced by the model's working memory (WM) capacity (cf. Daily, Lovett, & Reder, 2001). With a high WM capacity, the activation of the discourse referent that was

mentioned as the subject of the previous utterance remains high. This boost of activation implements the idea that the subject of the previous utterance is likely to be the current topic (Grosz et al., 1995; Stevenson et al., 1994). Thus, only when WM capacity is sufficient will grammatical function be used in determining the discourse topic.

A new empirical prediction following from our model is that adult listeners will show difficulties comprehending a topic shift if their WM capacity is limited. For example, if their WM is taxed by another task, they will be less likely to use the grammatical function of the referents in the discourse to determine the discourse topic. Rather, they will solely rely on the frequency and recency of the referents.

Experiment

Using a dual-task experiment, we have investigated the effect of additional WM load on the interpretation of pronouns in different discourse contexts. Participants had to memorize a sequence of either three (low WM load condition) or six digits (high WM load condition) for recall at the end of the trial. While memorizing the digits, participants had to read short stories with or without topic shift (indicated by new or same subject). The final sentence of the stories started with an ambiguous pronoun. The story was followed by a comprehension question to elicit the referent of the ambiguous pronoun.

The data of 52 participants was analyzed. As predicted, WM load affected adults' interpretation of subject pronouns in stories with a topic shift (Figure 1): with high WM load adults less often selected the subject of the previous sentence as referent of the pronoun, but more often selected the firstly introduced referent (which was also more frequently mentioned). No significant effect of WM load was found in the stories without a topic shift, where the firstly introduced character was the subject in all sentences.

These results support the prediction following from our cognitive model that the interpretation of pronouns in discourse is dependent on the amount of WM capacity available for interpretation.

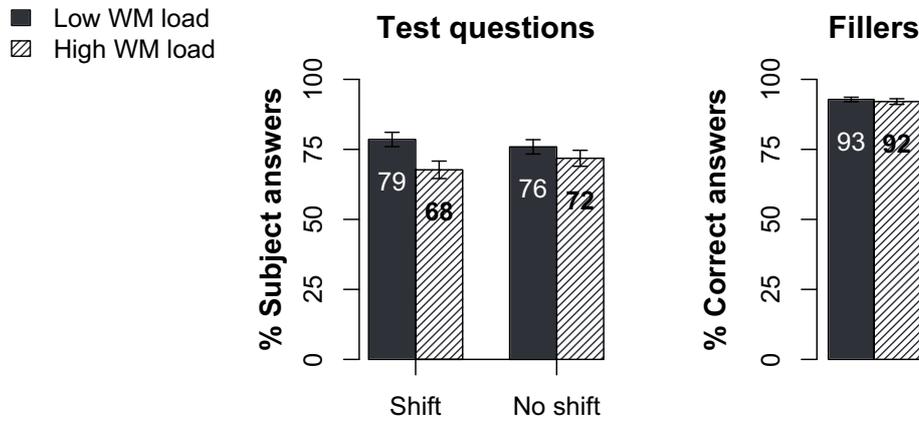


Figure 1: The percentage (\pm SE) that the previous subject was selected as referent of the pronoun (left) and the percentage correct answers on the filler questions (right) following stories with and without topic shift. The filled bars show the answers in the low WM load condition, the striped bars show the answers in the high WM load condition..

References

- Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* New York: Oxford University Press, USA.
- Daily, L. Z., Lovett, M. C., & Reder, L. M. (2001). Modeling individual differences in working memory performance: A source activation account. *Cognitive Science*, 25(3), 315.
- Grosz, B. J., Weinstein, S., & Joshi, A. K. (1995). Centering: a framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2), 203-225.
- Koster, C., Hoeks, J., & Hendriks, P. (2011). Comprehension and production of subject pronouns: Evidence for the asymmetry of grammar. In A. Grimm, A. Müller, C. Hamann & E. Ruigendijk (Eds.), *Production-comprehension asymmetries in child language*. Berlin: De Gruyter.
- Stevenson, R. J., Crawley, R. A., & Kleinman, D. (1994). Thematic roles, focus and the representation of events. *Language and Cognitive Processes*, 9(4), 519-548..

An ACT-R Model of Credibility Judgment of Micro-blogging Web Pages

Q. Vera Liao (liao28@illinois.edu)

Department of Computer Science, University of Illinois, 201 N. Goodwin Ave
Urbana, IL 61801 USA

Peter Pirolli (pirolli@parc.com)

Palo Alto Research Center, 3333 Coyote Hill Rd.
Palo Alto, CA 94304

Wai-Tat Fu (wfu@illinois.edu)

Department of Computer Science, University of Illinois, 201 N. Goodwin Ave
Urbana, IL 61801 USA

Abstract

In this paper, we propose an ACT-R cognitive model for making credibility judgments about the credibility of Twitter authors. We abstracted the cognitive processes involved in three levels: attending to information on Web page, comprehending information to identify credibility cues, and integrating credibility cues to make a judgment. We represent basic knowledge required for making credibility judgment using declarative memory in ACT-R which is seeded with experiences of Twitter messages that have been passed through a Latent Dirichlet Allocation topic modeling process. Comparisons of model credibility judgments to human credibility judgments from controlled experiments show weak to strong correlations that range from $r = 0.31$ to $r = 0.83$ depending on the specific task.

Keywords: Web credibility judgment, ACT-R

When people make credibility judgments about Web-based content and its sources, people must perceive, comprehend and deliberate on the merits and flaws of available cues to make the judgment. Complexity arises from the fact that the judgment is rarely based on a single cue, but requires the integration of multiple cues. These cues may interact with or contradict each other, and accumulate over the course of interaction with the Web content. We present a cognitive modeling approach to investigate multi-cue Web credibility judgment.

Cognitive models have been applied to explain and predict human interaction with Web-based content, primarily focusing on relevance-based browsing or search. For example, MESA (Miller & Remington, 2004) and SNIF-ACT (Fu & Pirolli, 2007) are models that simulate how users navigate through websites to search for information relevant to a given task. Web credibility judgment is a complex high-level cognitive process that may be highly dependent on the goal of the user. Therefore, instead of building a universal model, our goal is to propose a framework that can be easily modified for different contexts, and demonstrate it with a specific task. In this study, we attempt to build an ACT-R model of credibility judgment when processing Twitter micro-blogging content.

Website credibility models are often conceptualized along two dimensions. One dimension, represented by stage

models (Wathen & Burkell, 2002), focuses on the iterative process of credibility evaluation, i.e., how the assessment takes place when users open a page, read the contents, and are further involved with the site. The other dimension, following a bottom-up approach, seeks to examine what elements on a Web page, and to what extent, impact users' credibility judgments. Detailed cognitive models have the potential to model the iterative processes of stage models and the impact of specific Web cues in different task and content contexts.

We chose to analyze a task with simplified Twitter page, which allows us to ignore the complex interactions between multiple types of information cues but focus on the iterative process of attending to, processing and evaluating information on a Web page. This study was also motivated by the potential value of building predictive models for evaluating information credibility of micro-blogging, and more broadly, user generated contents on Internet.

In the following section, we will first introduce the modeling task and a preliminary study conducted with the task. Conclusions drawn from the preliminary study are incorporated into the ACT-R model. In the second part we will describe the ACT-R model. Lastly, we will present a model validated by human data from a second experiment with the same credibility judgment tasks.

Modeling Task and Preliminary Study

The modeling task was based on a Twitter study conducted by Canini et al.(2011). Twitter is the popular micro-blogging service that enables users to add text-based posts of up to 140 characters, known as "tweets", on their own page. The goal of the study was to explore what factors on a Twitter page may impact users' credibility judgment about the Twitter author. Understanding this process is important because it may help improve the design of micro-blogger recommendation systems and user interfaces to help users to discover credible sources and content.

In the Canini et al. (2011) experiment, participants were presented with a page generated to represent individual Twitter users. Each of these pages included a user name and icon, a set of social status statistics (number of following, followers and tweets), 40 latest tweets by the user, and a

word cloud summarizing all his/her previously generated content (Figure 1). Among other things, each participant was asked to rate presented Twitter users' credibility in making judgments in the specific domain of car purchases. Three variables were manipulated in Canini et al (2011) in constructing the Twitter pages representing users:

- (1) *Content domain.* The top 10 experts suggested in the WeFollow directories of *car*, *investing*, *wine*, *fantasy football*, *dating* plus 10 random accounts were selected. WeFollow is a popular Twitter user recommendation system. It has topic directories such as car, football, etc, where users can sign up if they are experts or interested in the topic. Wefollow ranks all users based how many users in the same directory are following him/her. Experts from the car domain were considered *on-topic* with respect to the target task of judging recommendations for car purchases, the other domains were *cross-topic*.
- (2) *Social status.* For each page, the social status was randomly set to be high or low. For a high social status, the presented user had a large number of following/followers (more than 1000) and a large number of tweets (more than 100).
- (3) *Visualization.* The page was randomly set to be tweets only, word cloud+ tweets, and word cloud only.

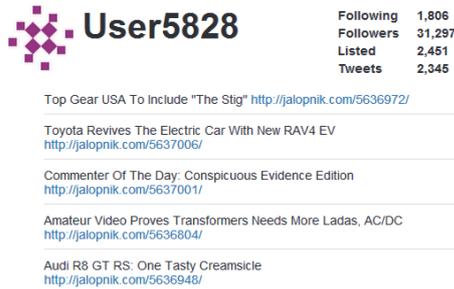


Figure 1. Modeling Task Interface in the tweets only condition, which is used for modeling task

The Canini et al (2011) results showed that the directory from which the Twitter author was selected had strong influence on perceived credibility. Not surprisingly, those selected from the car directory (on-topic) led to significantly higher credibility ratings than those from other directories (cross-topic). It was also found that users considered someone who talked a lot about dating were the least credible in giving car price suggestion, while experts in investing had a credibility rating in between the dating and car directories, possibly because the task of suggesting car price is related to financial decisions. It was also found that social status and visualization factors had smaller but statistically significant influences on credibility judgment.

We built an ACT-R model for this credibility judgment task. The credibility ratings given by the model are positively influenced by on-topic contents and negatively influenced by certain cross-topic contents. The model also has the capacity to process other contextual features on the Web page, such as social status.

Model Framework

We now present the general framework of the cognitive model for Web credibility judgment, and how this is implemented in ACT-R. Representations of knowledge are stored in declarative and procedural memory modules in ACT-R. Declarative memory, consisting of facts is represented by memory chunks built into the model. Procedural memory, representing knowledge about how we do things is represented as productions.

As shown in Figure 2, the model framework assumes a process consisting of three phases. First, the model attends to information on the page. The first phase includes processes that mostly involve attention and perception, such as fixing attention on tweets and initiating reading. For the ACT-R model, by attending to a tweet, e.g., "happy driving and car shopping", the model will recognize the word "happy", "driving", "car" and "shopping" by making use of its vocabulary knowledge in declarative memory.

In the second phase, the model comprehends information it has attended to, which leads to the identification of information cues that may potentially impact the credibility judgment. We use the spreading activation mechanism of ACT-R to implement this process. Retrieval of each chunk in declarative memory in ACT-R is determined by a chunk's activation. Activation reflects the degree to which a chunk is likely to be needed or relevant in the current context. The chunk with highest activation and above a set threshold is most likely to be retrieved. In addition to the base level activation which reflects the prior use of the chunk itself, the chunk will also receive activation spread from related chunks currently attended by the model. For example, when the model reads the tweet "happy driving and car shopping", each of the word spreads activation to potentially related topics. Both the word "car" and "driving" spread activation to the "car" topic, making its activation higher than other topics, e.g., "shop", which only receives activation from the word "shopping". Then the topic "car" will be retrieved, as being identified to be the topic of this particular tweet. Optionally, this phase may also involve inferences made based on the perception of other features on the Website. For example, if the model reads a large number of followers, it may identify it as a cue of high social status.

In the third phase, the model will deliberate on the information cues it identified and integrated them to make a credibility judgment. In the ACT-R model, we use the *blending mechanism* (Lebiere, 2005) to implement this phase. When using blending, if there are multiple candidate chunks satisfying the retrieval request specification but with different values in certain slots, the model will construct a same type of chunk containing slot values that "blend" over those multiple values. More specifically, ACT-R will retrieve a chunk that contains a compromise value, V , in the target slot that is determined by:

$$V = \text{Min} \sum_i P_i (1 - \text{Sim}(V, V_i))^2$$

where V_i is the value held in the target slot of the existing chunks i . P_i is the probability of retrieving existing chunk i ,

which is determined by the activation of chunk i . When making a credibility judgment, we assume that the model utilizes knowledge of previously stored instances of credibility judgments, i.e., prior knowledge that a certain cue is an indication of being credible or non-credible, and strength of that indication varies. The model blends all the instances it retrieves based on cues identified from the Web page to make the judgment. For example, the model will identify that topics concerned with “car”, “gas” and “dating” are discussed in the tweets. It will then decide that mentioning of “car” related information is a strong indicator of credibility for giving car price suggestion, which is represented by a strong activation spread from chunk “car” to chunk “credible”. Similarly, it may decide mentioning of “gas” related information is a less strong indicator of credibility, while mentioning of “dating” related information may be an indicator of non-credibility and thus spread activation to the chunk “non-credible”. The model will integrate the credibility indications of all cues according to the total activation received by the credible chunk and non-credible chunk to make the credibility judgment.

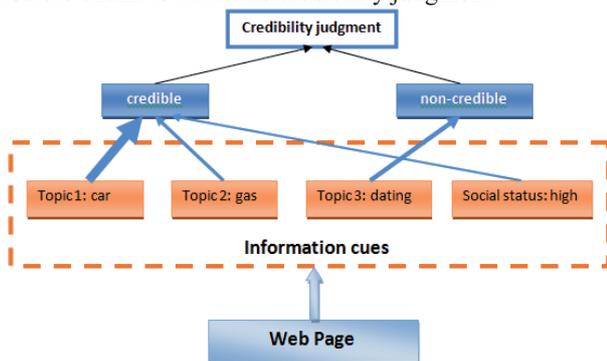


Figure 2. Model Framework

ACT-R Model for Twitter Author Credibility Judgment

The ACT-R model for Twitter page credibility judgment uses two buffers in addition to the basic ACT-R buffers: a word buffer and a credibility cue buffer. The content of the word buffer reflects the text that the model attends to and holds in a short-term memory. The credibility cue buffer contains cues identified by the model which may potentially have impact on credibility judgment. In the following section we will describe how we construct the declarative and procedural memory to work with the two buffers.

Declarative Memory

The declarative memory of this ACT-R includes word chunks, topic chunks and credibility chunks, and optionally, contextual cue chunks. Because Web credibility judgment process may involve frequent use of declarative knowledge, it is important to build declarative memory that allows adequate knowledge for such process. Therefore, to enable the model to process Twitter pages, we built a corpus by collecting all retrievable tweets from 1800 individual

Twitter accounts (maximum=3000 tweets each) randomly chosen from different WeFollow directories, and constructed the declarative memory from this large dataset.

Word Chunk

We identified the 3000 stemmed words (which are not stop words such as a, the, of, etc) with the highest frequency from the Tweets corpus. Word chunks to represent each of the 3000 words were added into the declarative memory. These represent the vocabulary knowledge the model has to process the Twitter contents.

Topic Chunk

We used Latent Dirichlet Allocation (LDA) topic modeling (Blei et al., 2003) to identify topics that can be used to comprehend Twitter message content. LDA is a generative model which posits that a document, i.e., the collection of observed words, is a mixture of unobserved topics and that each word’s creation is attributed to one or several of the document’s topics. We exploited an LDA topic model produced in Canini et al. (2011) that used documents constructed by aggregating all the tweets in the same corpus as described above. Following Canini et al. (2011), we selected 500 topics with the highest frequency to be the topic chunks in declarative memory. They represent the knowledge for processing and comprehending Tweets. Each word chunk is associated with one or multiple topics.

Contextual Information Cue Chunk

All the contextual information cues, if any, could be added into declarative memory as contextual cue chunks. For example, to process social status in the task, we could add “high social status chunk” and “low social status chunk” into the declarative memory.

Credibility Chunk

We built two credibility chunks, a “credible” chunk and a “non-credible” chunk which have a value slot to represent the two extreme values (rating 1 and rating 7) of credibility judgment ratings. Each credibility cue chunk (including topic chunk and contextual information cue chunk) is associated with either the credible chunk or non-credible chunk, and the strength of association varies.

Procedural Memory

The procedural memory was built to execute the credibility judgment process as shown in Table 1. The model will start by reading the textual content in sequence (i.e., from left to right, top to bottom). When the model attends to a word, and it has a corresponding word chunk in the declarative memory, the chunk will be retrieved and placed in the word buffer. With the limitation of short term memory, only a limited number of words will be stored in the buffer. When the word buffer reaches its capacity, if a new word chunk is retrieved, the earliest word attended will be removed, and each existing cue in the buffer will be moved to the earlier slot. Hence the model will iteratively hold the latest words it attends to in the word buffer.

When processing the contents, the model attempts to identify topics based on what it has just read. At any

moment, the word buffer contains a list of words. Each of the word chunks is associated with one or multiple topic chunks in the declarative memory. All these words will collectively decide the strength of association spreading to the topic chunks. The topic that is above retrieval threshold and receives highest activation will be placed into the credibility cue buffer. Since the list of words in the word buffer will continuously change, the model may identify multiple topics as the model reads through the page. For the current model, we only allow topics that are not currently in the credibility cue buffer to be retrieved. Optionally, the credibility cue buffer has slots to hold contextual credibility cues. Similar to the word buffer, the credibility cue buffer also has limited number of slots, and will only keep the latest credibility cues.

Resembling human behavior, the model may stop before it finishes processing all information. Anytime the model identifies a new credibility cue, it chooses between the production that halts further reading and a production to continuing processing. In ACT-R, when there are multiple productions waiting to be fired, the chances that production i will be fired is decided by:

$$P(i) = \frac{e^{U_i/\sqrt{2s}}}{\sum_j e^{U_j/\sqrt{2s}}}$$

where U_i represents the utility value set for production i and s is a utility noise parameter. We set the utility of the production for continuing processing to be higher than the production to halt reading. Therefore at different points of processing the Web content, the model has chance to stop, but the chance is still lower than that of continuing reading.

When either the model chooses to stop or it reaches the end of the page, the production for making the credibility judgment will be fired. As discussed in the previous section, there is a credible chunk with a rating slot of value 7, and a non-credible chunk with a rating slot of value 1. They receive activation spread from the credibility cue buffer, as positive credibility cues are associated with the credible chunk, and negative ones are associated with the non-credible chunk. The model uses the blending mechanism to blend the rating values of credibility chunk and non-credibility chunk based on the activation of the two chunks.

Table 1. Model Procedural

Attend to word IF there is corresponding chunk in declarative memory THEN push the chunk into word buffer	IF NOT THEN attend to next word
Hold word in word buffer IF there is open slot in word buffer THEN hold the word chunk in the latest open slot	IF NOT THEN remove the earliest word and move each word chunk to an earlier slot to open the latest slot
Understand topic IF there is topic(s) above retrieve threshold & the topic(s) is not held in the credibility cue buffer THEN retrieve a topic	IF NOT THEN attend to next word
Hold topic in credibility cue buffer IF there is open slot in credibility cue	IF NOT

buffer THEN hold the topic in credibility cue buffer	THEN remove the earliest cue and move each cue to an earlier slot to open up the latest slot
Decide to stop of continue IF stop production is fired THEN start to make credibility judgment	IF NOT THEN attend to next word
Make credibility judgment IF model stops reading or no more content left for processing THEN make credibility judgment blending credibility chunks	

Strength of Association

ACT-R calculates the activation of each chunk by:

$$A_i = B_i + \sum_k \sum_j W_{kj} S_{ji} + \varepsilon$$

B_i is the base-level activation, which reflects the recency and frequency of practice of chunk i . The component $W_{kj} S_{ji}$ reflects spreading of activation from retrieved chunks to related chunks in the declarative memory. S represents the strength of association. W can be set to decide the weighting of different slots in a buffer to spread activation to the declarative memory. ε is the system noise value.

There are two phases in the model where the activation spreading plays a role: 1) the emergence of topic is determined by the collective activation spread from the words held in word buffer, and 2) the activation of credibility chunks is determined by the collective activation spread from the credibility cues held in the credibility cue buffer. We will describe the rules we used to set the strength of spreading activation below.

Strength of association from word to topic

By using the LDA topic model for the tweets corpus described above, we calculate the strength of association from word to topic by:

$$S_{wt} = \log(P(w|t) / P(w))$$

where $P(w|t)$ is the LDA-estimated probability of word w given the occurrence of topic t and $P(w)$ is an estimated of the probability of word occurrence.

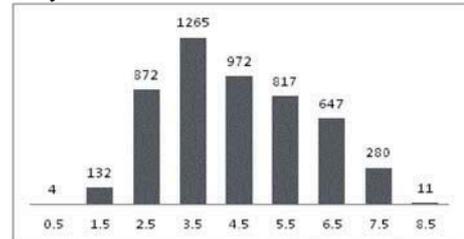


Figure 3. Distribution of strength of associations from word to topic

For the model, we set the limit of number of word slots for each topic chunk to be 10. It means we only identify the strength of association of the top 10 words for each topic, and overall we identified 5000 strength of associations (10 for each of the 500 topics). The distribution of the strength of association (number of associations falling in each range of strength) is shown in Figure 3. This approach enables the model to have the knowledge to infer the potential explanations (i.e., topics) of each word that it attends to.

Strength of association from credibility cue to credibility

Strength of association from topic chunks to credibility chunks indicates the extent to which the particular topic is regarded as an indicator of credibility or non-credibility by the model. The model reads the task description and attends to key words of the task (e.g., for the car price suggestion task, the key words are “car” and “price”). For each of the key words, the model attempts to identify topics that are highly related to the key words. We set the current model to select the top 30 topics, with which the attended key word chunk has the highest strengths of association. Then the model increases the strength of association from the topic to credibility chunk by the same amount of strength. It allows the model to use a bottom up approach to identify topics that are associated with the task goal and that may have positive impact on credibility judgment.

According to the results of our preliminary study, there seemed to be topics with negative effects on the credibility rating (e.g., dating related topics). While it is difficult to exhaustively identify all the negatively associated topics, since we only intend to test the model with directories of car, dating and investing at the current stage, we manually selected a few topics that are strongly associated with words frequently used by authors in dating directory (e.g., dating, sex, etc), and set strength of associations from these negative topics to the non-credible chunk.

Similarly, contextual cue chunks in the credibility cue buffer, if any, will spread activation to either of the two credibility chunks. For example, the high social status chunk, if held in credibility cue buffer, will spread activation to credible chunk.

Pilot Validation

We used the same setup and procedure as in the Canini et al. (2011) experiment, which asks participants to rate a Twitter author’s credibility for giving car price suggestions. However, instead of manipulating multiple features on the page, we focused on only users’ tweet contents. We selected the latest 40 tweets from the top 10 users recommended in the WeFollow directories for cars, investing and dating. We recruited $N = 7$ participants to complete the credibility rating task. Each participant judged all the 30 pages in random order.

We first performed a repeated measure ANOVA on participants’ credibility ratings, with author domain (car,

dating, investing) as the independent variable. The result showed that the main effects of directory is significant ($F(2,12)=4.82, p=0.03$), meaning credibility ratings given to the authors from the three directories are different. Post-hoc analysis showed that the ratings given to authors from car directory are significantly higher than those from dating directory ($F(1,6)=12.05, p=0.01$). The model results showed the same pattern. As the model results may vary if it stops reading at different parts of the page, we ran the model for 10 times and calculated the mean ratings for each page. We performed t-test between each pair of author directories for the mean rating of each page given by the model. It shows the ratings given to Twitter author selected from car directory are significantly higher than those from dating directory ($t(18)=5.46, p<0.01$), and those from investing directory ($t(18)=4.62, p<0.01$). The results suggest that, the model, like human participants, is able to infer the source credibility for the task goal (i.e., car price suggestion) based on the micro-blogging content created by the person.

We are aware that the perceived credibility varies even for Twitter authors selected from the same directory. For example, some car experts may not necessarily talk about cars in their tweets, while others may tweet about it frequently. Potentially, one practical use of a cognitive model for Web credibility judgment is the capability of predicting perceived credibility for individual pages. We therefore looked into the correlations between human judgment and model judgment for individual pages. Specifically, we expect the model to be able to differentiate higher credibility from lower credibility Twitter sources as judged by humans.

Figure 4 shows the human results and model results for credibility ratings about 10 users chosen from the WeFollow directories of cars, investing and dating. The fit for all the 30 pages between human and model results is $R^2=0.69$. The fit for the 10 authors from car directory is $R^2=0.56$, correlation for investing directory is $R^2=0.30$, correlation for dating directory is $R^2=0.10$. Although the results do not show a good fit for investing and dating directory, we are aware that the current model may not be able to exhaustively identify information cues that negatively affect credibility judgments.

At a broader level of analysis we tested to what extent the model could predict the valence (i.e., low vs high) of the credibility judgment. To this end, for the 30 pages with authors from the car, dating and investing directories, we

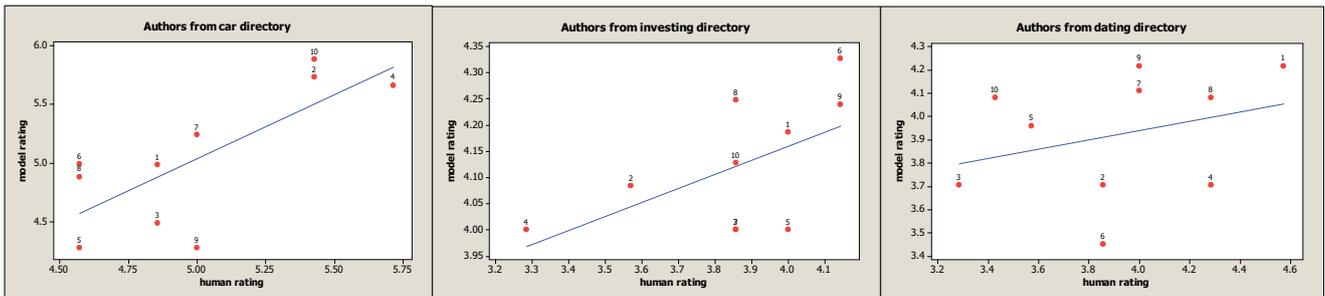


Figure 4. Human and model results

performed a median split analysis of Twitter user credibility rating. Each Twitter user was coded as being (1) high-credibility or low-credibility based on whether it was above median or below median in terms of average human rating and (2) high-credibility or low-credibility based on whether the Twitter user was above median or below median on model ratings. The results showed that, for 26 out of 30 pages, human results and model results fall into the same bucket (with the exception of 2 high-credibility and 2 low-credibility pages). The Chi-square test on this 2×2 median split showed they are dependant ($\chi^2=16.13, p<0.01$).

We further tested the prediction of valence within each author directory. We performed the same median split analysis for the 10 pages with authors from car directory. The results showed that, for 8 out of 10 pages, human results and model results fall into the same bucket (with the exception of 1 high and 1 low credibility page, $\chi^2=5.33, p=0.02$). To further verify these pages are perceived to have different valence of credibility, we performed repeated measure ANOVA with human ratings for the 8 pages which fall in same bucket for both human and model results, with the valence (high/low) as independent variable. It shows the ratings are significantly different ($F(1,6)=10.52, p=0.02$). We performed the same analysis for authors selected from investing directory. We also found, for 8 out of 10 pages, human ratings and model ratings fall into the same high or low bucket (with the exception of 1 high and 1 low credibility page, $\chi^2=5.33, p=0.02$). The ANOVA verified the ratings given to the two groups of pages is marginally significant ($F(1,6)=4.52, p=0.07$). We did not look into the dating directory because of the lack of knowledge about negative cues as discussed earlier. These results proved that the model was able to predict the valence of credibility for individual pages.

Discussion

In this study, we proposed a framework for a cognitive model for making credibility judgments of Web content or its sources, and implemented it in ACT-R. We exploited Twitter content to induce an LDA topic model that was used to seed declarative memory and support an instance-based judgment process based on the ACT-R blending mechanism. In general, the model is able to infer the level of credibility of Twitter authors by differentiating authors with on-topic content for the task goal and those without. It is also able to predict the perceived credibility of individual users with on-topic contents.

The model performs three phases of cognitive process to make a credibility judgment of Web content or sources: attending to information on the page, comprehending the information to infer credibility cues, and making credibility judgment by integrating these credibility cues. During the comprehending phase, the spreading activation mechanism of ACT-R is used to identify the most likely explanation when there are multiple pieces of observed information and each may have multiple explanations. The blending mechanism is used to generate a judgment by integrating

credibility cues, each of which may indicate a different level of credibility. Although we built the model with a Twitter author judgment task in this paper, by changing the model knowledge for processing information on a Web page, and knowledge about credibility of different cues, the model could be modified to apply to different media, content, or sources.

The major limitation of current model is its lack of complete knowledge about the credibility indications of various information cues, especially those that may negatively impact credibility judgments. Future research is needed to explore this research question.

Acknowledgements

This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes not withstanding any copyright notation here on.

Reference

- Anderson, J. R. (1983). A spreading activation theory of memory. *Journal of Verbal Learning and Verbal Behavior*, 22, 261-29
- Anderson, J. R. & Lebiere, C. (1998). The atomic components of thought. Mahwah, NJ: Erlbaum
- Blei, D. M., Ng, A., Jordan, M. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3 (4-5), 993-1022.
- Canini, K. R., Suh, B., & Pirolli, P. L. (2011). Finding credible information sources in social networks based on content and social structure. *In Proceedings of the third IEEE International Conference on Social Computing (SocialCom)*.
- Fu, W.-T., Pirolli, P. (2007), SNIF-ACT: A Model of Information-Seeking Behavior in the World Wide Web. *Human-Computer Interaction*, 22, 355-412.
- Lebiere, C. (2005). Constrained functionality: Application of the ACT-R cognitive architecture to the AMBR modeling comparison. In K. A. Gluck & R. Pew (Eds.), *Modeling human behavior with integrated cognitive architectures*, 63-112. Mahwah, NJ: Erlbaum.
- Lebiere, C. (1999). A blending process for aggregate retrievals. *In Proceedings of the 6th ACT-R Workshop*. George Mason University, Fairfax, Va.
- Miller, C. S., & Remington, R. W. (2004). Modeling information navigation: Implications for information architecture. *Human Computer Interaction*, 19, 225-271
- Wathen, C.N. and J. Burkell(2002), Believe it or not: Factors influencing credibility on the Web. *Journal of the American Society for Information Science and Technology*, 2, 134-144.

Computationally Efficient Forgetting via Base-Level Activation

Nate Derbinsky (nlderbin@umich.edu)

John E. Laird (laird@umich.edu)

University of Michigan, 2260 Hayward Street
Ann Arbor, MI 48109-2121 USA

Keywords: large-scale cognitive modeling

Introduction

As we apply cognitive models to complex, temporally extended tasks, removing declarative knowledge from memory, or *forgetting*, will become important both to model human behavior, as well as to scale computationally. The *base-level activation* (BLA) model predicts that the availability of specific memories is sensitive to frequency and recency of use. Memory decay based on this model has long been a core commitment of the ACT-R theory (Anderson et al., 2004), as it has been shown to account for a class of memory retrieval errors (Anderson, Reder, & Lebiere, 1996), and has been used in Soar (Laird, 2012) to investigate task-performance effects of forgetting short-term (Chong, 2003) and procedural (Chong, 2004) knowledge. Prior work has addressed many of the computational challenges associated with retrieving a single memory according to the BLA model (Petrov, 2006; Derbinsky, Laird, & Smith, 2010; Derbinsky & Laird, 2011). However, efficiently removing items from memory, while preserving BLA-model fidelity, is a different problem, which we address here. We formally describe the computational problem; present a novel approach to forget according to BLA in large memories; and evaluate using synthetic data.

Problem Formulation

Let memory M be a set of elements, $\{m_1, m_2, \dots\}$. Let each element m_i be defined as a set of pairs (a_{ij}, k_{ij}) , where k_{ij} refers to the number of times element m_i was activated at time a_{ij} . We assume $|m_i| \leq c$: the number of activation events for any element is bounded. These assumptions are consistent with the ACT-R declarative memory when bounding chunk-history size (Petrov, 2006). This is also consistent with the semantic memory in Soar (Laird, 2012).

We assume that activation of an element m at time t is computed according to the BLA model (Anderson et al. 2004), where d is a fixed decay parameter:

$$B(m, t, d) = \ln\left(\sum_{j=1}^{|m|} k_j \cdot [t - a_j]^{-d}\right)$$

We define an element as *decayed*, with respect to a threshold parameter θ if $B(m, t, d) < \theta$. Given a static element m , we define L as the fewest number of time steps required for the element to decay, relative to time step t :

$$L(m, t, d, \theta) := \inf\{t_d \in \mathbb{N} : B(m, t + t_d, d) < \theta\}$$

For example, element $x = \{(3, 1), (5, 2)\}$ was activated once at time step three and twice at time step five. Assuming decay rate 0.5 and threshold -2, x has activation about 0.649 at time step 7 and is not decayed: $L(x, 7, 0.5, -2) = 489$.

During a model time step t , the following actions can occur with respect to memory M :

- S1. A new element is added to M .
- S2. An existing element is removed from M .
- S3. An existing element is activated y times.

If S3 occurs with respect to element m_i , a new pair (t, y) is added to m_i . To maintain a bounded history size, if $|m_i| > c$, the pair with smallest a (i.e. the oldest) is removed from m_i .

Thus, given a memory M , we define that the *forgetting* problem, at each time step, t , is to identify the subset of elements, $D \subseteq M$, that have decayed since the last time step.

Efficient Approach

Given this problem definition, a naïve approach is to determine the decay status of each element every time step. This test requires computation $O(|M|)$, scaling linearly with average memory size. The computation expended upon each element, m_i , will be linear in the number of time steps where $m_i \in M$, estimated as $O(L)$ for a static element.

Our approach draws inspiration from the work of Nuxoll, Laird, and James (2004): rather than checking memory elements for decay status, “predict” the future time step when the element will decay. First, at each time step, examine elements that either (S1) weren’t previously in the memory or (S3) were activated. The number of items requiring inspection is bounded by the total number of elements ($|M|$), but may be a small subset. For each of these elements, predict the time of future decay (discussed shortly) and add the element to a map, where the map key is the predicted time step and the value is a set of elements predicted to decay at that time. If the element was already within the map (S3), remove it from its old location before adding to its new location. All insertions/removals require time at most logarithmic in the number of distinct decay time steps, which is bounded by the total number of elements ($|M|$). At any time step, the set D is those elements in the set indexed by the current time step that are decayed.

To predict element decay, we perform a novel, two-phase process. After a new activation (S3), we first employ an *approximation* that is guaranteed to underestimate the true value of L . If, at a future time step, we encounter the element in D and it has not decayed, we then compute the exact prediction using a binary parameter search.

We approximate L for an element m as the sum of L for each independent pair $(a, k) \in m$. Here we derive the closed-form calculation: given a single element pair at time t , we solve for t_p , the future time of element decay...

$$\ln(k \cdot [t_p + (t - a)]^{-d}) = \theta$$

$$\ln(k) - d \cdot \ln(t_p + (t - a)) = \theta$$

$$t_p = e^{\frac{\theta - \ln(k)}{-d}} - (t - a)$$

Since k refers to a single time point, a , we rewrite the summed terms as a product. Furthermore, we time shift the decay term by the difference between the current time step, t , and that of the element pair, a , thereby predicting L .

Computing this approximation for a single pair takes constant time (and common values can be cached). Overall approximation computation is linear in the number of pairs, which is bounded by c , and therefore $O(1)$. The computation required for binary parameter search of an element is $O(\log_2 L)$. However, this computation is only necessary if the element has not decayed, or removed from M .

Evaluation

This approach has been empirically evaluated for long-term tasks in the procedural and working memories of Soar (Derbinsky & Laird, 2012). In this paper, we focus on the quality and efficiency of our prediction approach and utilize synthetic data. Our data set comprises 50,000 memory elements, each with a randomly generated pair set. The size of each element was randomly selected from between 1 and 10, the number of activations per pair (k) was randomly selected between 1 and 10, and the time of each pair (a) was randomly selected between 1 and 999. We verified that each element had a valid history with respect to time step 1000, meaning that each element would not have decayed before $t=1000$. Also, each element contained a pair with at least one access at time point 999, which simulated a fresh activation (S3). For all synthetic experiments we used decay rate $d=0.8$ and threshold $\theta=-1.6$. Given these constraints, the largest possible value of L for an element is 3332.

We first evaluate the quality of the decay approximation. In Figure 1, the y-axis is the cumulative proportion of the elements and the x-axis plots absolute temporal error of the approximation, where a value of 0 indicates that the approximation was correct, and non-zero indicates how many time steps the approximation under-predicted. We see that the approximation was correct for over 60% of the elements, but did underestimate over 500 time steps for 20% of the elements and over 1000 time steps for 1% of the elements. Under the constraints of this data set, it is possible for this approximation to underestimate up to 2084 time steps. We also compared the prediction time, in microseconds, of the approximation to an exact calculation using binary parameter search. The maximum computation time across the data set was >19x faster for the approximation (1.37 vs. 26.28 $\mu\text{sec}/\text{element}$) and the average time was >15x faster (0.31 vs. 4.73 $\mu\text{sec}/\text{element}$).

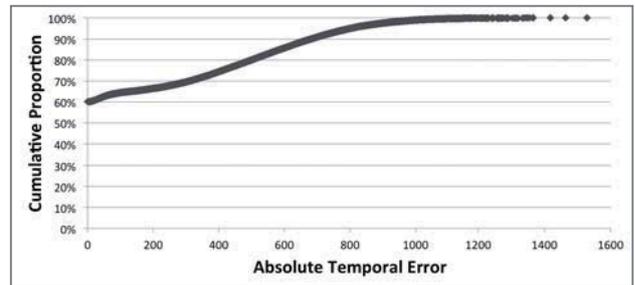


Figure 1. Evaluation of decay-approximation quality.

We did not compare these results with a naïve approach, as results would depend upon a model of memory size ($|M|$).

In conclusion, we presented a novel, two-phase forgetting approach that maintains fidelity to the base-level activation model and scales to large memories. The experimental results show that the first phase is a high-quality approximation and is an order of magnitude less costly than the exact calculation in the second phase.

Acknowledgments

We acknowledge the funding support of the Air Force Office of Scientific Research, contract FA2386-10-1-4127.

References

- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., Qin, Y. (2004). An Integrated Theory of the Mind. *Psychological Review*, 111 (4), 1036-1060.
- Anderson, J. R., Reder, L., Lebiere, C. (1996). Working Memory: Activation Limitations on Retrieval. *Cognitive Psychology*, 30, 221-256.
- Chong, R. (2003). The Addition of an Activation and Decay Mechanism to the Soar Architecture. *Proc. of the 5th Intl. Conf. on Cognitive Modeling* (pp. 45-50).
- Chong, R. (2004). Architectural Explorations for Modeling Procedural Skill Decay. *Proc. of the 6th Intl. Conf. on Cognitive Modeling*.
- Derbinsky, N., Laird, J. E. (2011). A Functional Analysis of Historical Memory Retrieval Bias in the Word Sense Disambiguation Task. *Proc. of the 25th AAAI Intl. Conf. on Artificial Intelligence* (pp. 663-668).
- Derbinsky, N., Laird, J. E. (2012). Competence-Preserving Retention of Learned Knowledge in Soar's Working and Procedural Memories. *Proc. of the 11th Intl. Conf. on Cognitive Modeling*.
- Derbinsky, N., Laird, J. E., Smith, B. (2010). Towards Efficiently Supporting Large Symbolic Declarative Memories. *Proc. of the 10th Intl. Conf. on Cognitive Modeling* (pp. 49-54).
- Laird, J. E. (2012). *The Soar Cognitive Architecture*, MIT Press.
- Nuxoll, A., Laird, J. E., James, M. (2004). Comprehensive Working Memory Activation in Soar. *Proc. of the 6th Intl. Conf. on Cognitive Modeling* (pp. 226-230).
- Petrov, A. (2006). Computationally Efficient Approximation of the Base-Level Learning Equation in ACT-R. *Proc. of the 7th Intl. Conf. on Cognitive Modeling* (pp. 391-392).

A Change for the Better? Assessing the Computational Cost of Re-representation

Todd Wareham (harold@mun.ca) and Robert Robere (robere.it@gmail.com)

Department of Computer Science, Memorial University of Newfoundland
St. John's, NL Canada A1B 3X5

Iris van Rooij (i.vanrooij@donders.ru.nl)

Donders Institute for Brain, Cognition, and Behaviour
Radboud University Nijmegen, Nijmegen, The Netherlands

Abstract

The ability to re-represent information—i.e., to see things in new ways—is essential for human reasoning, creativity, and learning. It forms the foundation of insight problem solving and scientific explanation, and is hypothesized to play a pivotal role in concept development in children. Re-representation is useful because it allows a cognizer to make sense of things in ways that were previously impossible. Yet, invoking this operation can quickly become computationally intractable in light of the combinatorial explosion of re-representation options to consider. Although this intractability may explain why discovering useful ways of re-representing information can be cognitively challenging at times (as in insight puzzles and creativity), it seems difficult to reconcile with automatic and apparently effortless forms of re-representation (as in everyday analogizing and children's development of concepts). To get more insight into the conditions that can make re-representation tractable, we performed computational complexity analyses of a formal model of re-representation as invoked in analogy derivation. We will discuss how our complexity results can help explain when and why re-representation can be invoked effectively and efficiently.

Keywords: Re-representation; Analogy; Structure Mapping; Computational Complexity; Fixed-Parameter Tractability

Introduction

Many theories of cognitive abilities operate on mental representations of information, each of which assumes a particular encoding of relevant situations and concepts. As there are typically many possible encodings, one's initial representation may in fact be inappropriate for the task at hand, slowing down or even stopping the ability from functioning. In such cases, it is hypothesized that humans change encodings and re-represent stored information.

Re-representation has been invoked in many cognitive theories. It allows natural analogies that rely on semantic rather than syntactic matching (e.g., “Bob running into the cave” is like “Alice walking into a room” \Rightarrow “Bob moves into the cave” is like “Alice moves into the room” (Gentner & Kurtz, 2006)). The powerful abstractions generated by such re-representation in turn have been hypothesized to underlie certain types of concept development in children (e.g., the emergence of abstract relations and attributes: “X is hotter than Y” \Rightarrow “temperature(X) is greater than temperature(Y)” (Gentner, Rattermann, Markman, & Kotovosky, 1995)). More radical types of re-representation can in turn lead to totally new ways of envisioning particular situations and concepts, and thus have been invoked in theories of insight problem solving (Ohlsson, 1992), scientific discovery (Gentner et al., 1997), and creativity (Welling, 2007).

Investigating re-representation experimentally is difficult, but there is increasing evidence for its psychological reality (Gentner & Kurtz, 2006; Kurtz, 2006). This has motivated the development of computational theories of re-representation (e.g., Ohlsson (1992); Yan, Forbus, and Gentner (2003); Krumnack, Gust, Kühnberger, and Schwering (2008)), which has raised the following conundrum: The combinatorial explosion of re-representation options that must be considered in such theories seems to be computationally intractable. This intractability may explain why certain cognitive activities invoking re-representation like insight problem solving and creativity are challenging, but is at odds with how other forms of re-representation invoked in everyday reasoning or cognitive development seem so effortless and automatic.

In this paper, we assess the computational difficulty of a basic type of re-representation, namely individual predicate re-representation within Gentner's Structure Mapping Theory of analogy derivation (SMT) (Gentner, 1983; Yan et al., 2003). We give the first proof that such re-representation is computationally intractable, even when invoked in the context of incremental rather than general analogy derivation. This finding indicates that constraints on both the re-representation process and its inputs must be exploited to yield tractability. In the second part of this paper we illustrate a methodology suitable for identifying such constraints. We also discuss how our results can help explain when and why re-representation can be invoked effectively and efficiently.

Computational-level Models

Analogies are defined over concepts, which are represented in SMT by predicate-structures composed of entities (e.g., SUN, PLANET) and predicates relating those entities (as well as other predicates) (e.g., ATTRACTS(SUN, PLANET)). Predicate-structures are naturally represented as vertex-labelled directed acyclic graphs in which entities are leaves, predicates are internal vertices, and predicates are linked to their arguments by arcs (see part (a) of Figure 1).

An **analogy** “*T* is (like) a *B*”, where *B* and *T* are predicate-structures, is a mapping from a portion of *B* to a portion of *T* that satisfies the following three conditions:

1. The mapping is **structurally consistent**, i.e., matching relations must have matching arguments and any element in one predicate-structure matches at most one element in the other.

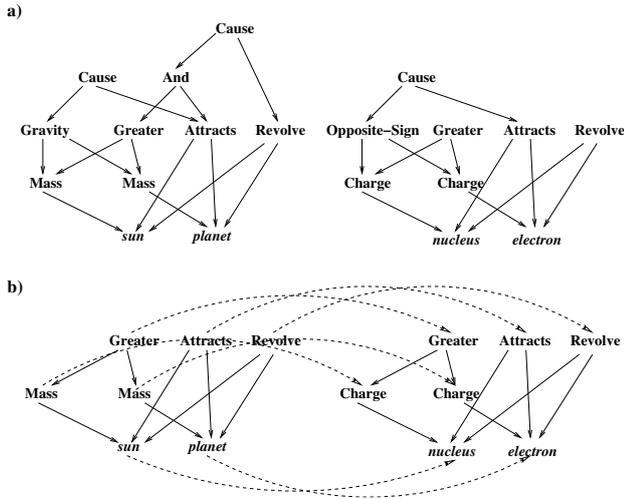


Figure 1: Analogy Derivation in Structure-Mapping Theory. (a) Two graph representations of predicate structures encoding descriptions of the solar system (left) and the Rutherford model of the atom (right). (b) An optimal analogy between the structures in (a), where dotted arrows indicate the mappings between corresponding pairs of predicates and objects.

2. **Relational focus:** The mapping must involve common predicates but need not involve common objects, *i.e.*, matched predicates must have the same type, argument, number and order but matched objects need not have the same name.
3. **Systematicity:** The mapping tends to match interconnected, deeply-nested predicate-substructures.

Let $val(A)$ be the systematicity of an analogy A . The most systematic analogy between a pair of predicate-structures is an **optimal analogy** (see part (b) of Figure 1).

Under SMT, re-representation of predicates is only invoked to better the analogical match between two given predicate-structures. As such, it relaxes identical-only predicate-type matches (*e.g.*, ATTRACTS \rightarrow ATTRACTS) to allow selected non-identical matches (*e.g.*, WALK \rightarrow MOVE). There are two classes of mechanisms for performing re-representations:

1. **Rule-guided** (part (a) of Figure 2): A predicate of type x can be re-represented as a predicate of type y if there is a rule $x \rightarrow y$. Collections of rules can be encoded as predicate-type similarity tables (represented explicitly (Holyoak & Thagard, 1989) or generated implicitly by predicate decomposition (Gentner et al., 1995)) or generalization lattices (in which the most specific predicate-types are at the bottom of the lattice and the most abstract predicate-types are at the top) (Winston, 1980).
2. **Context-guided** (part (b) of Figure 2): A predicate p of type x can be re-represented as a predicate of type y if p appears in a structural context immediately “outside” an

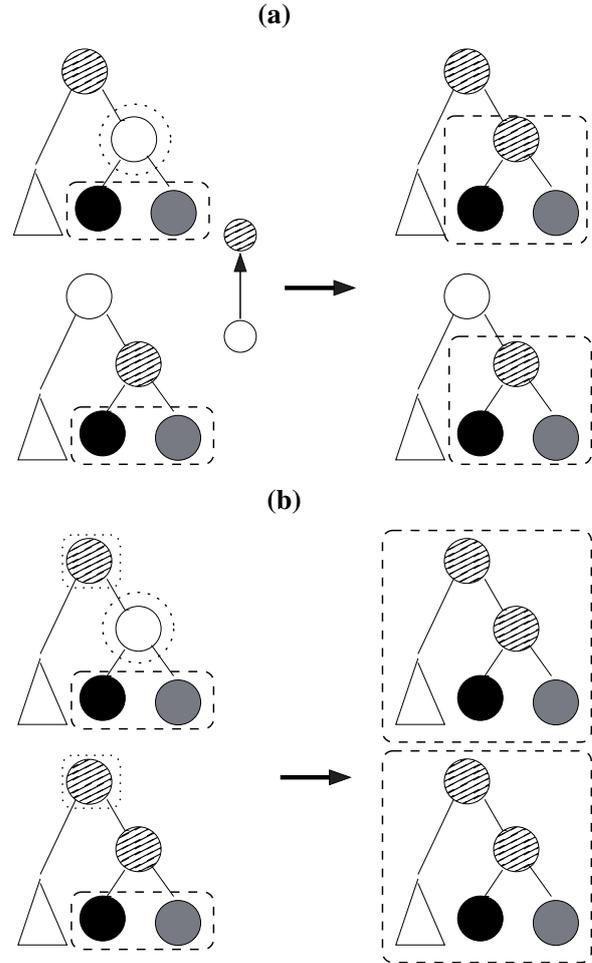


Figure 2: Re-Representation Mechanisms in Structure-Mapping Theory. (a) Rule-guided. (b) Context-guided. Analogically-matched regions are enclosed by dashed boxes.

existing analogy between two predicate-structures which, if p 's type is changed, will allow an incremental addition to that analogy which increases its systematicity. The most basic type of context is a “hole”, in which a pair of predicates in B and T have different types but both their arguments and parents have the same types and can be matched.

Analogy derivation alternates with such re-representation until a satisfactory analogy is reached. In any one round of re-representations, it is assumed that each predicate in the given predicate-structures can change at most once. Though we focus here on single-predicate re-representation, more complex re-representations involving larger changes in structure are also possible (Yan et al., 2003).

Acting on all available re-representation opportunities can both be computationally expensive and potentially result in analogies that are meaningless or misleading, *e.g.*, “analogical hallucinations” (Gentner & Kurtz, 2006, Page 616). There are many possible strategies for selecting which re-

representations to perform. Two general principles underlie all such strategies (Yan et al., 2003, Page 1269):

1. **Systematicity:** All else being equal, re-representation suggestions that lead to increases in the systematicity of the derived analogy will be preferred.
2. **High Selectivity:** The selection process should be tightly controlled, so that very few of the possible opportunities are selected for consideration.

The above considerations yield the following computational-level models of representation under SMT. All three models assume that re-representation is done to improve on a given optimal analogy. The first of these models is general, in that it does not require that the created analogy be an extension of the given analogy.

ANALOGY DERIVATION WITH RE-REPRESENTATION (ADR)

Input: Predicate-structures B and T , an optimal analogy $A(B, T)$, a rule-set R , and integers k and c .

Output: Predicate-structures B' and T' and an analogy $A'(B', T')$ such that (i) B' and T' are derivable from B and T by at most k applications of rules from R and (ii) $val(A') - val(A) \geq c$.

The second and third models are restrictions of the first, as they are required to extend the given analogy.

ANALOGY IMPROVEMENT WITH RULE-GUIDED RE-REPRESENTATION (AIR[R])

Input: Predicate-structures B and T , an optimal analogy $A(B, T)$, a rule-set R , and integers k and c .

Output: Predicate-structures B' and T' and an analogy $A'(B', T')$ such that (i) $A \subset A'$, (ii) B' and T' are derivable from B and T by at most k applications of rules from R , and (iii) $val(A') - val(A) \geq c$.

ANALOGY IMPROVEMENT WITH CONTEXT-GUIDED RE-REPRESENTATION (AIR[C])

Input: Predicate-structures B and T , an optimal analogy $A(B, T)$, and integers k and c .

Output: Predicate-structures B' and T' and an analogy $A'(B', T')$ such that (i) $A \subset A'$, (ii) B' and T' are derivable from B and T by at most k context-guided re-representations, and (iii) $val(A') - val(A) \geq c$.

For simplicity, we will assume that all context-guided re-representations in the third model are of the basic “hole” type shown in part (b) of Figure 2.

It is possible that the act of analogy derivation rather than re-representation may artificially boost the difficulty of the models described above. To this end, we will also analyze a fourth model of re-representation, whose goal is to re-represent a given predicate-structure in order to satisfy a polynomial-time computable function $Prop$ that returns either *True* or *False*, e.g., does the re-represented T contain a particular type of easily-recognizable structure?

GENERAL DERIVATION WITH RE-REPRESENTATION (GDR)

Input: Predicate-structure T such that $Prop(T) = False$, rule-set R , and integer k .

Output: Predicate-structure T' such that (i) T' derivable by at most k applications of rules from R and (ii) $Prop(T') = True$.

The four models above are those that will be considered below. However, as will be explained later in the paper, results derived relative to these models have implications for a broad range of cognitive theories invoking re-representation.

Re-representation is Intractable

To investigate the computational (in)tractability of the models of re-representation given in the previous section, we have adopted standard complexity-theoretic proof techniques from Computer Science (Garey & Johnson, 1979). Using these techniques, we have proven the following (see the supplementary materials for proofs¹):

Result 1 *ADR, AIR[R], AIR[C], and GDR are NP-hard.*

These results imply that there do not exist any algorithms for performing basic re-representation in the sense of the models considered here in polynomial time for all inputs (i.e., time upper-bounded by some function n^c where n is a measure of input size and c is some constant).² In other words, all algorithms for these models will run in exponential time or worse (i.e., time upper-bounded at best by some function c^n for c and n as above). As exponential-time algorithms have unrealistically long runtimes for all but very small inputs, they are generally considered to be computationally intractable (Garey & Johnson, 1979).

Given that it is NP-hard to derive analogies of a specified systematicity (van Rooij, Evans, Müller, Gedge, & Wareham, 2008; Veale & Keane, 1997), the NP-hardness of ADR is not unexpected. The NP-hardness of AIR[R] and AIR[C] is surprising, as deriving analogies that must be built on and include given analogies (Forbus, Ferguson, & Gentner, 1994) was not previously thought to be intractable. This suggests that the act of re-representation *all by itself* is intractable, which is confirmed by the NP-hardness of GDR. That all of these results hold in the most basic case as well – that is, re-representation of individual predicates — has additional power, as this means that these results may actually *under-estimate* the complexity of more complex types of re-representation invoking larger scale structural changes such as those proposed in Yan et al. (2003).

All this being said, the above does *not* say that re-representation is impossible – rather, it suggests that re-representation in practice may require one or more additional constraints on the inputs and/or the re-representation process

¹<http://www.cs.mun.ca/~harold/Papers/ICCM12supp.pdf>

²This assumes that the conjecture $P \neq NP$ is true, which is widely believed within the Computer Science community on both theoretical and empirical grounds (Fortnow, 2009).

not considered so far in order to be computationally practical. In the next section, we describe a methodology that can be used to both model such specific constraints and investigate their computational effects.

A Method for Identifying Tractability Conditions

A computational problem that is intractable for unrestricted inputs may yet be tractable for non-trivial restrictions on the input. This insight is based on the observation that some *NP*-hard problems can be solved by algorithms whose running time is polynomial in the overall input size and non-polynomial only in some aspects of the input called **parameters**. In other words, the main part of the input contributes to the overall complexity in a “good” way, whereas only the parameters contribute to the overall complexity in a “bad” way. In such cases, the problem Π is said to be **fixed-parameter tractable** (Downey & Fellows, 1999) for that set of parameters. The following definition states this idea more formally.

Definition 1 *Let Π be a problem with parameters k_1, k_2, \dots . Then Π is said to be fixed-parameter (fp-) tractable for parameter-set $K = \{k_1, k_2, \dots, k_c\}$ if there exists at least one algorithm that solves Π for any input of size n in time $f(k_1, k_2, \dots, k_c)n^c$, where $f(\cdot)$ is an arbitrary function and c is a constant. If no such algorithm exists then Π is said to be fixed-parameter (fp-) intractable for parameter-set K .*

In other words, a problem Π is fp-tractable for a parameter-set K if all superpolynomial-time complexity in solving Π can be confined to the parameters in K . In this sense the unbounded nature of the parameters in K can be seen as a reason for the intractability of the unconstrained version of Π . For any given fixed-parameter (in)tractability result, other results may be implied courtesy of the following lemmas:

Lemma 1 *If Π is fp-tractable for K then Π is fp-tractable for any K' such that $K \subset K'$.*

Lemma 2 *If Π is fp-intractable for K then Π is fp-intractable for any K' such that $K' \subset K$.*

It follows from the definition of fp-tractability that if an intractable problem Π is fp-tractable for parameter-set K , then Π can be efficiently solved even for large inputs, provided only that all the parameters in K are relatively small. In the next section we report on our investigation of whether or not parameters may be used in this way to render the models *ADR*, *AIR[R]*, *AIR[C]*, and *GDR* tractable.

What Does (and Doesn't) Make Re-representation Tractable?

Table 1 lists the parameters that we will consider in our fixed-parameter analyses of re-representation. Each of these parameters is of interest for different reasons. Parameters o and p are already known to individually render analogy derivation fp-tractable (van Rooij et al., 2008; Wareham, Evans,

Table 1: Overview of Parameters Considered.

Name	Description
o	Maximum number of objects over B and T
p	Maximum number of predicates over B and T
k	Amount of allowed re-representation
$ R $	Rule-set size
a	Total number of re-representation opportunities in B and T

& van Rooij, 2011) and may in turn make analogy derivation with re-representation fp-tractable. Parameters k and $|R|$ explicitly and implicitly, respectively, encode the High Selectivity principle for re-representation selection strategies, and should thus be small in practice. Finally, in addition to considering parameters that separately characterize the inputs (o, p) and the re-representation process ($k, |R|$), we will investigate parameter a , which in a sense encodes the degree of interaction between the given predicate-structures and the re-representation mechanisms (either rules in R or hole-contexts) in terms of the number of opportunities that these inputs provide for the application of these mechanisms.

The results of our analyses relative to these parameters are given below (see the supplementary materials for proofs). As we are still in the early stages of our investigation, these results in tandem with Lemmas 1 and 2 do not yet fully characterize the parameterized complexity of our models relative to all possible combinations of the considered parameters. However, even at this initial stage, we can still draw some interesting conclusions and conjectures.

Let us start with the fp-intractability results:

Result 2 **ADR* and *GDR* are fp-intractable for parameter-sets $\{o, k, a\}$ and $\{k, |R|\}$.*

Result 3 **AIR[R]* is fp-intractable for parameter-set $\{o, k, a\}$.*

Result 4 **AIR[C]* is fp-intractable for parameter-set $\{o, k\}$.*

Though there are still some open questions (in particular, the parameterized status of *AIR[R]* relative to $\{|R|\}$, *AIR[C]* relative to $\{a\}$, and *GDR* relative to $\{p\}$), these results in tandem with Lemma 1 establish that almost none of the parameters considered here can, if individually restricted to small values, render any of our models computationally feasible. The same also holds for any combinations of the parameters within the parameter-sets mentioned in these results. Of particular note is the fact that neither of the four models considered here can be made feasible by restricting k alone. This suggests that other principles in addition to High Selectivity must underlie re-representation selection strategies if re-representation is to be made feasible. These principles may have to be model-specific; for example, the current scarcity of fp-intractability results for *AIR[R]* and *AIR[C]* suggests that requiring derived analogies to build on given analogies may provide model-specific opportunities for restrictions that yield fp-tractability.

Consider now the fp-tractability results:

Result 5 *ADR, AIR[R], and AIR[C] are fp-tractable for parameter-set $\{p\}$.*

Result 6 *GDR is fp-tractable for parameter-set $\{p, |R|\}$.*

Result 7 *ADR and AIR[R] are fp-tractable for parameter-set $\{o, |R|, a\}$.*

Result 8 *AIR[C] is fp-tractable for parameter-set $\{o, a\}$.*

Result 9 *GDR is fp-tractable for parameter-set $\{|R|, a\}$.*

Each of these results implies that if all parameters in that result’s parameter-set have small value, then the model mentioned in that result can be computationally feasible on inputs of arbitrary size. For example, Result 8 says that if o and a are simultaneously of small value, then AIR[C] may be computationally feasible. Results 7, 8, and 9 are of particular interest. The constraint on predicate-structure size imposed by o is not overly onerous, as many kinds of predicate-structures are based on a relatively small number of objects (Schlimm, 2008); moreover, it seems reasonable to conjecture that for certain applications (*e.g.*, those involving large-scale re-representation rules), a and $|R|$ may be suitably small.

Generality of Results

All of the intractability results reported in this paper, though defined relative to a specific theory of analogy derivation, have broad applicability. This is because the models examined here are restricted versions of models for other cognitive theories that invoke re-representation, *e.g.*,

- The re-representation modes encoded in our models are used in many cognitive theories (*e.g.*, GDR’s single-structure re-representation parallels re-representation in insight problem solving (Ohlsson, 1992)).
- The predicate-structures on which our models are based are a powerful but basic form of representation, and it seems reasonable to conjecture that these other theories can be phrased in terms of predicate-structures.
- The basic single-predicate-change rules and hole-contexts used in our models are special cases of the more complex re-representation invoked in these other theories.

Results for models of other theories that satisfy the above then follow from the well-known observation that intractability results for a problem Π also hold for any problem Π' that has Π as a special case and can hence solve Π (suppose Π is intractable; if Π' is tractable, then it can be used to solve Π efficiently, which contradicts the intractability of Π – hence, Π' must also be intractable).

Our fp-tractability results are more fragile, as innocuous changes in the form of the inputs or the re-representation rules and contexts may in fact violate assumptions critical to the operation of the algorithms underlying these results. For now, we can say that as the parameters mentioned in Results 7, 8,

and 9 encode only the combinatorics of re-representation possibilities (via $|R|$ and/or a) and require only that the structures generated by each such possible set of re-representations can be evaluated to determine if they comprise a viable solution in a reasonable amount of time, these results apply to all models whose input-types and re-representation mechanisms satisfy these conditions.

Discussion

Our research was motivated by the question of how the computational difficulty of re-representation in general can be reconciled with the ease of many instances of everyday re-representation. To address this question, we first set out to assess using formal methods whether re-representation as proposed in one such instance, namely analogy derivation, was computationally tractable. We found that this is not the case. In contrast, even these models of analogy derivation that only allow the simplest forms of re-representation can be proven *NP*-hard (Result 1). This means that no practical (read: polynomial time) algorithm can exist that perform such re-representation for all representations. To our knowledge, this is the first formal proof of the intractability of re-representation in the context of analogy derivation.

As this left the questions of how and under what conditions re-representation can become tractable, we performed complexity analyses to identify parameters that when restricted to small values render re-representation tractable (see Table 1 and its associated section for results). We believe that the following two of our findings are of particular interest:

1. Limiting the amount of re-representation (*i.e.*, small k) does not by itself (nor when combined with many other parameters) make re-representation tractable (Results 2–4).
2. What does make re-representation tractable in the case of analogy (and, as noted above, many other more complex models) is when all of the parameters in the sets $\{p\}$ (Result 5) or $\{o, |R|, a\}$ (Results 7 and 8) are simultaneously restricted to take small values.

The latter set in (2) may be applicable to re-representation in everyday analogy derivation (especially those cases applying large-scale re-representations) and the former set may be reasonable for re-representation in concept development, as it is strongly hypothesized that children’s representations are object- and attribute-rich and relationally poor (*i.e.*, small p) (Gentner et al., 1995). The question now is whether these properties actually hold in these and other observedly fast forms of re-representation. If empirical evidence of these properties can be found, then our tractability results provide a psychologically plausible explanation of how the modelled forms of re-representation can be tractable despite the intractability of re-representation in general.

To summarize, in this paper we have given the first formal proofs not only that re-representation is computationally difficult even by itself, but that there are restrictions that

may allow it to operate quickly in practice. Promising directions for future work include extending parameterized analyses of the models defined here to other parameters (in particular, parameters like a that describe interactions between the given input and the re-representation process), developing good fixed-parameter algorithms for re-representation within analogy derivation for implementation in large-scale AI systems like the Companions architecture (Forbus & Hinrichs, 2006), and investigating in detail the extent to which results and conclusions presented here apply to other models of re-representation-assisted analogy such as AMBR (Kokinov & Petrov, 2000) and HDTP (Krumnack et al., 2008) as well as models of insight problem solving and creativity.

Acknowledgments

The authors would like to thank three anonymous reviewers for comments that improved the presentation of this paper. TW was supported by NSERC Discovery Grant 228104.

References

- Downey, R., & Fellows, M. (1999). *Parameterized Complexity*. Berlin: Springer.
- Forbus, K. D., Ferguson, R. W., & Gentner, D. (1994). Incremental structure-mapping. In *Proceedings of the 16th Annual Conference of the Cognitive Science Society* (pp. 313–318). Mahweh, NJ: Lawrence Erlbaum Associates.
- Forbus, K. D., & Hinrichs, T. (2006). Companion Cognitive Systems: A Step Towards Human-level AI. *AI Magazine*, 27(2), 83–95.
- Fortnow, L. (2009). The Status of the P Versus NP Problem. *Communications of the ACM*, 52(9), 78–86.
- Garey, M. R., & Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: W.H. Freeman.
- Gentner, D. (1983). Structure-Mapping: A Theoretical Framework for Analogy. *Cognitive Science*, 7, 155–170.
- Gentner, D., Brem, S., Ferguson, R., Wolff, P., Markman, A. B., & Forbus, K. D. (1997). Analogy and creativity in the works of Johannes Kepler. In T. B. Ward, S. M. Smith, & J. Vaid (Eds.), *Creative thought: An investigation of conceptual structures and processes* (pp. 403–459). Washington, DC: American Psychological Association.
- Gentner, D., & Kurtz, K. J. (2006). Relations, Objects, and the Composition of Analogies. *Cognitive Science*, 30, 609–642.
- Gentner, D., Rattermann, M. J., Markman, A., & Kotvosky, L. (1995). Two forces in the development of relational similarity. In T. J. Simon & G. S. Halford (Eds.), *Developing cognitive competence: New approaches to process modelling* (pp. 263–313). Mahweh, NJ: Lawrence Erlbaum Associates.
- Holyoak, K. J., & Thagard, P. (1989). Analogical mapping by constraint satisfaction. *Cognitive Science*, 13, 295–355.
- Kokinov, B. N., & Petrov, A. (2000). Dynamic Extension of Episode Representation in Analogy-Making in AMBR. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society* (pp. 274–279). Mahweh, NJ: Lawrence Erlbaum Associates.
- Krumnack, U., Gust, H., Kühnberger, K.-U., & Schwering, A. (2008). Re-representation in a Logic-Based Model for Analogy Making. In W. Wobcke & M. Zhang (Eds.), *AI 2008* (pp. 42–48). Berlin: Springer-Verlag.
- Kurtz, K. J. (2006). Re-representation in comparison: building an empirical case. *Journal of Experimental & Theoretical Artificial Intelligence*, 17(4), 447–459.
- Ohlsson, S. (1992). Information-processing explanations of insight and related phenomena. In M. Keane & K. Gilhooly (Eds.), *Advances in the Psychology of Thinking* (pp. 1–44). London: Harvester-Wheatsheaf.
- Schlimm, D. (2008). Two Ways of Analogy: Extending the Study of Analogies to Mathematical Domains. *Philosophy of Science*, 75(2), 178–200.
- van Rooij, I., Evans, P., Müller, M., Gedge, J., & Wareham, T. (2008). Identifying Sources of Intractability in Cognitive Models: An Illustration using Analogical Structure Mapping. In *Proceedings of the 30th Annual Conference of the Cognitive Science Society* (pp. 915–920). Austin, TX: Cognitive Science Society.
- Veale, T., & Keane, M. T. (1997). The Competence of Sub-Optimal Theories of Structure Mapping on Hard Analogies. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI'97)* (Vol. 1, pp. 232–237). San Mateo, CA: Morgan Kaufmann.
- Wareham, T., Evans, P., & van Rooij, I. (2011). What Does (and Doesn't) Make Analogical Problem Solving Easy? A Complexity-Theoretic Investigation. *Journal of Problem Solving*, 3(2), 30–71.
- Welling, H. (2007). Four Mental Operations in Creative Cognition: The Importance of Abstraction. *Journal of Creativity Research*, 19(2–3), 163–177.
- Winston, P. H. (1980). Learning and reasoning by analogy. *Communications of the ACM*, 23(12), 689–703.
- Yan, J., Forbus, K. D., & Gentner, D. (2003). A theory of rerepresentation in analogical matching. In *Proceedings of the 25th Annual Conference of the Cognitive Science Society* (pp. 1265–1270). Mahweh, NJ: Lawrence Erlbaum Associates.

Defining Factors of Interest for Large-scale Socio-cognitive Simulations

Ryan Kaulakis, Changkun Zhao, Jonathan H. Morgan, Jeremiah W. Hiam, Joseph P. Sanford, Frank E. Ritter
(rmk216@psu.edu, cuz111@psu.edu, jhm5001@psu.edu, jwh189@psu.edu, jps314@psu.edu, frank.ritter@psu.edu)

The College of Information Sciences and Technology
The Pennsylvania State University

Abstract

We examine how several environmental factors influence cognition and the emergence of social networks in artificial societies. Using a large-scale socio-cognitive simulation (VIPER), we generated social networks consisting of 20, 40, and 60 agents. We tested the impact that environmental factors such as population size, map configuration, and run time, particularly on the formation of social ties in memory. We analyzed 1,080 ego-nets across 27 conditions, measuring the number of links and average degree of each network. While all these factors influenced these network measures, our results suggest that population size has the largest influence. In addition, we examined what impact activation values and retention parameters have on the construction of social networks in memory, finding that a shift in activation values resulted in a loss of links, thus indicating a cognitive foundation for Dunbar's Number (Dunbar, 1998) for the maximum number of social ties stored in memory.

Keywords: ACT-R; socio-cognitive network; memory; network formation

Introduction

We seek to show two things here: first, that several commonly used parameters in network science have an impact and need to be reported, but which are currently under-reported; and, secondly, that cognitive architectures used in network science will have different results than intelligent agents lacking cognitive plausibility. So, in this paper, we examine three common environmental factors that influence the construction of social networks: population size, environmental configuration, and run duration. Later work uses these results to examine how these factors influence specific network relations such as Dunbar's Number (Dunbar, 1998) in (Zhao, Kaulakis, Morgan, Hiam, Sanford, et al., 2012; Zhao, Kaulakis, Morgan, Hiam, & Ritter, 2012).

This work is motivated by a desire to better understand how socio-cognitive processes influence the development of persistent patterns of relations, represented in this paper as network topologies. By socio-cognitive processes, we refer to both those cognitive resources and mechanisms necessary to create and sustain social ties, as well as those group-level factors known to moderate human decision-making as modeled by (Morgan, Morgan, & Ritter, 2010). We, also examine memory retention because it seems foundational to understanding the construction and maintenance of social networks within cognitive architectures.

We refer to these networks as socio-cognitive networks. We differentiate socio-cognitive networks from other social networks by their means of interaction, or the degree to which the networks structure, modes of communication, and resources derive from an external medium (e.g., cell phone networks or web-based friend networks). We believe a deeper

understanding of the constraints imposed by memory decay on network formation is important because such an understanding may allow us to refine our predictions regarding the likely structures and capabilities of socio-cognitive networks. In particular, we believe an agent-based approach can allow us to deepen our understanding of the relationship between the carrying capacity (in this case an agents ability to recall its friends network) of a networks nodes and its topology.

To explore these questions, we introduce a set of cognitive models and experiments that vary across three factors. The outputs of this model are ideal networks (whole networks representing the total number of agent interactions that occurred within a single run) and ego-nets (declarative representations of the agents friends network). For any one run, there is then a single ideal network and as many ego-nets (networks from egocentric points of view) as there are agents. We tested the effects of environmental conditions on the construction of the agents ego-nets by comparing multiple ideal networks with their related ego-nets. We also analyzed to what extent these factors (coupled with memory constraints) influenced the constructed networks, measured by differences in the number of links and average node degree.

Our model is unusual in that we model social processes using a full cognitive architecture (ACT-R). To our knowledge, (Carley, 1991, 1992; Carley & Newell, 1994) were the first to implement social network models using a cognitive architecture (Plural Soar) to study organizations. More recently, (Gonzalez, Lerch, & Lebiere, 2003), (Lebiere, Gonzalez, Dutt, & Warwick, 2009), (Reitter & Lebiere, 2010), and (Juvina, Lebiere, Martin, & Gonzalez, 2011) have used cognitive architectures to model human decision making in collaborative tasks. While our work builds upon these efforts, we focus here on the formation of social networks. Further, while cognitive architectures bring great power, they are computationally expensive. We thus, must address both questions of utility and theoretical subsumption. In other words, what do these architectures uniquely bring to the table that we need? In the next section, we will address these questions by orienting ourselves with respect to past work in social modeling before describing our model more fully.

Computational Social Models

We briefly describe a cross-section of related work. We will move from simple non-cognitive models to models that center on cognition and cognitive modeling. All of these approaches are agent-based, or refer to predictions based on individual decision-making.

In many models, the role of actors is often described us-

ing closed-form mathematical formulas that offer parsimony, but often prove difficult when modeling complex stochastic systems. In contrast, computational models typically try to model the effects of a wider range interrelated factors using more complex but bounded actors (Axelrod & Hammond, 2003). These effects often include both environmental, as well as cognitive factors.

Drawing from work in environmental and social psychology (Kraut, Fussell, Brennan, & Siegel, 2002; Allen, 1977), we found that actor proximity fundamentally influenced the evolution of network topologies by determining the interaction frequencies of actors across the network. Allen (Allen, 1977) demonstrated that the probability of two people communicating in an environment could be defined by a decreasing hyperbolic function of the distance between them. After a certain distance, the probability that two people will communicate decreases rapidly, making link formation unlikely. We thus choose to focus on factors that directly affect agent proximity or inter-agent distance: population size, run duration, and map configuration.

We expect that larger populations acting over longer periods in fully connected rooms will result in the richest declarative network structures. We also expect that layouts that afford greater distances will result in interaction networks that consist of multiple components, leading to smaller ego-nets. We also expect that map configurations characterized by nexus points will exhibit behaviors similar to the water-cooler effect. We, however, are less certain where we might see thresholds in network formation, where for instance population growth no longer has an effect or run time is no longer relevant. We expect these thresholds will provide us a better understanding of the cognitive dimension behind the shifts in group behavior associated with changes in group size.

Drawing from previous work in cognitive science (Simon, 1984; Prietula & Carley, 2001), we were interested in how bounded rationality influenced network formation by constraining network construction in memory. We believe that these constraints will result in interesting and sometimes unexpected aggregate behaviors that may provide insights for future efforts in multi-level modeling. In particular, we believe the concept of nodal carrying capacity, or the number of agents any one agent can retain in its social declarative representation, may be helpful for predicting the capabilities and structure of a network of interest. To that end, we examine how shifts in activation values and retention parameters, as well as differences in environmental factors contribute to the consolidation and retention of social ties in memory. This concept is similar to those of Dunbar (Dunbar, 1998) and the Bernard-Killworth median (McCarty, Killworth, Bernard, Johnsen, & Shelley, 2001). Dunbar's number arises from the limitations associated with the neocortex. Because maintaining a stable relationships requires repeated memory activations in the human neocortex to identify not only one-on-one relationships but also third party relationships (i.e., the knowledge that my friend is also friends with other actors

who I, in some senses, monitor), the cognitive load associated with maintaining this set of relationships in memory rises exponentially as group size increases (Dunbar, 1998, p.63). Based on retrospective empirical studies, (Dunbar, 1998, p.65–78) argues that this ratio between cognitive load and group size underlies the small-world effect observed by Milgram and others.

Nodal carrying capacity: The effect of agents memory and space

Thus far, we have categorized the factors that influence the construction of social networks into two groups: factors that influence interaction frequency and factors that influence retention. In this section, we will discuss each of these categories, and give a general prediction on how the factors associated with each effect the construction and retention of cognitive ego-nets.

Interaction Frequency

We examine three factors that influence the interaction frequencies associated with a given social network. These factors include: population size, duration of contact (run time), and environment map connectivity (defined by a grid ratio, or the number of links over the total number of links possible in a similar grid).

Population size: We suspect that population density is the most influential factor governing interaction frequency. Because we are comparing map configurations consisting of the same number of rooms, we manipulate population density by testing three different population sizes (20, 40, and 60 agents).

Length of simulation: The time period that agents interact directly influences the structure of the simulated social network because more time allows for more interaction opportunities, making it more likely that agents will establish a stable network. Consequently, determining the run times necessary for a network to reach a stable state under a given set of conditions is important for accurately representing the formation of a group of interest. We use total degree with respect to time as a measure of network stability. Modeling memory decay, however, seems essential for determining a meaningful notion of network stability. Otherwise, we suspect simulated networks will tend to achieve arbitrary and inflated levels of connectivity ending with complete connectivity at infinite time.

Environment configuration: We believe that the configuration of the rooms of the environment influences the structure of the simulated social network. We measure the relative connectivity of our three map configurations by defining its grid ratio. The grid ratio is the ratio of the number of edges over the total number of edges possible for a rectangular grid containing the same number of rooms.

We tested three map configurations. The first configuration is a full 5x5 grid with grid ratio 1.0. We expect this environment will result in relatively high connectivity. The second

configuration (shown in Figure 1a) is a two-hallway configuration with grid ratio 0.6. This configuration should lead to low connectivity due to the large distances between agents. The third configuration (shown in Figure 1b) has a central area with grid ratio 0.75. We believe this central meeting point will lead to agent behavior between 1 and 2.

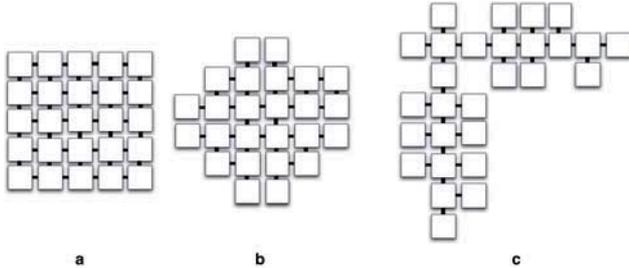


Figure 1: a: Full Grid, b: Central Area, c: Hallway

Cognitive factors

To better simulate the construction of social networks, it is necessary to consider the behavior patterns of agents at the cognitive level. In this paper, we particularly focus on memory decay. We examine memory's effect on tie formation by using Anderson's activation theory (Anderson et al., 2004) to model the construction of social knowledge in declarative memory. In our model, the number of friends depends on the number and size of active long-term memory chunks associated with the agent's social relationships. The number of active memory chunks can be influenced by several factors, including initial memory activation, retrieval threshold, memory decay speed, time of retrieval, and practice time.

Experiment Environment

To model multi-agent social behavior using cognitive architectures, we constructed a simulation environment called VIPER.

The VIPER Server

VIPER is a lightweight, extensible, text-based simulation environment. It uses the telnet protocol to handle plain text communication between agents and the simulation environment and forces a separation of environment and agent. VIPER records agent behaviors within the environment using a logging system that provides a detailed list of every action taken by all agents chronologically ordered to the tenth of a second. VIPER resolves events in either real or accelerated time: the network's speed and frequency of communication is determined by its component agents with no queue of events being enforced within the environment.

Within VIPER, agents are situated in maps of interconnected rooms. In each room, the agents can see and communicate locally. Agents can walk freely around the rooms, and can interact with objects in their environment.

To connect ACT-R to VIPER, we implemented a VIPER client, the Telnet Agent Wrapper (TAWA) for ACT-R, in

Common Lisp. It handles everything from logging in, waiting for synchronization, logging, halting, and writing results to CSV files automatically. Additionally, it provides functions to examine the environment, speak, listen, move, and otherwise control virtual bodies in VIPER.

When an ACT-R model is wrapped by TAWA, any execution of model code is delayed until a privileged administrator actor inside of VIPER signals synchronization. Additionally, any error conditions are caught by TAWA and used to return standard UNIX error codes instead of dropping into the debugger. For example, a successful run returns zero to the parent process, while any error (e.g., network errors like the server being unreachable) causes a non-zero return value. Returning error codes allows automated error checking in large scale experiments.

All of our experiments were conducted on a 2GHz eight-core Linux 2.6.31 under Ubuntu 11.04 server with 8GB of RAM, with SBCL 1.0.52 as our Lisp run-time. We use ACT-R 6 described in (Anderson et al., 2004).

Synchronization

Because memory decay and networks are strongly temporal, we paid special attention to time. When designing our experiments, we developed VIPER and TAWA to use a synchronization process. During an experimental run, TAWA delays the evaluation of the model code until synchronization, this means that no agent experiences time before the synchronization signal. Further, all ACT-R models are set to run in real-time, and for the full time, using the run-full-time function from standard ACT-R with `:real-time` enabled. All agents run for the same amount of real time, so they all halt after the same perceived period after starting. Thus, the total time experienced is the same for all agents.

Scalability

Early benchmarks showed that ACT-R processes took up about 80MB per process. Because we had only 8GB of RAM, we would only have been able to run about 100 processes on a single machine before swapping. To reduce the per-process footprint, a number of optimizations were implemented. Basic space reductions were achieved by using the DECLARE operator, as well as by compiling all libraries, removing the debugger, and saving the whole system (sans the ACT-R agent model) as a system image. This reduced our per-process memory footprint somewhat, but they were not the biggest contributions towards memory usage reduction.

In SBCL version 1.0.52, the `-merge-core-pages` flag was recently added. This flag enables Kernel SamePage Merging (Arcangeli, Eidus, & Wright, 2009) under recent versions of Linux. This optimization flags shared areas of memory as merge-able unless modified. Because a significant percentage of our agents were replicated, we could reduce the per-process memory footprint as low as 8MB per process in benchmarks. Thus, the only things that increase the size of this footprint during run-time are changes to memory done

by the ACT-R agent model. Benchmarks have shown reasonable performance with 700 test ACT-R agents.

ACT-R Agents

We built an ACT-R model to conduct a *random walk* in VIPER. The model contains two declarative memory types: a goal type containing current location, remaining steps, total friends counts; and a friend type to store a friends name. The model consists of four basic components with 9 productions. First, the *walking* component selects an available direction randomly, and moves itself through VIPER. Second, the *waiting* component utilizes temporal buffer of ACT-R to wait two real-time seconds when the agent comes into a new room, to simulate how long it might take a normal person to enter a room. Third, the *checking* component (with 3 productions) checks if the current room is empty. Finally, the *memorizing* component (with 3 productions) will check its declarative memory first to check if it can recall the fact that the agent it has encountered is a friend. If it is not, the model will create a new friend chunk, and store the encountered agents name in it using the imaginal buffer.

Experiment and Results

In this section, we discuss the results of our analysis of 27 system logs and 1,080 ego-net logs.

Experiment Parameters

In the first experiment, we have 27 runs of our simulation to test three environmental factors: population size, run time, and map configuration. To manipulate room configuration, we use a 5x5 grid map and two other maps shown in Figure 1. All parameters are shown in the Table 1.

Table 1: Experiment Parameters

Factors	Testing Values
Population Size	20, 40, 60
Running time (seconds)	125, 250, 500
Map Configuration (grid ratio)	Full Grid (1.00), Central (0.75), Hall (0.60)

To test cognitive parameter, we make all agents output the activation value of each friend chunks as a ego-net log file, which contains friend name and location of the last meeting. In ACT-R, the activation value represents the memory retention of an object or an event. With the activation value of each relation chunk, we could easily find the weight of each friend tie in memory.

Results

As noted in section 4, our simulation generates two types of network data: log data extracted from Viper directly, and egocentric data stored in the agents declarative memory. In this section, we will present samples of the data and some related network measures.

Ideal Network Figure 2 shows a sample interaction-network. The nodes in the figure are the agents in the simulation (20 in total); the links in the figure are unweighted, only representing the co-occurrence of two agents in a given room during the run.

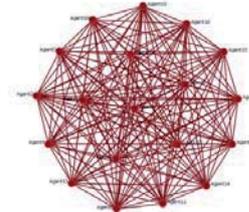


Figure 2: The network of log data for 20 agents running 125s in the Hallway.

Table 2 shows the measure comparison between 7 runs. We found that the population size and running time influence the network measures, reflected in Average Node Degree and Degree Centrality. We find that the Total Links and Average Degree increase when population size increases. Total Links and Average Standard Degree also increase when doubling run-time.

Table 2: Measurements of seven sample runs, grouped by variable (italicized); N is Population Size, T is Run-time in seconds, R is Grid Ratio, Links is Total Links, Degree is Average Node Degree, Centrality is Degree Centrality.

N	T	R	Links	Degree	Centrality
20	125	0.60	324	0.436	0.627
40	125	0.60	1410	0.784	0.227
60	125	0.60	3801	0.370	0.651
20	<i>125</i>	0.60	324	0.436	0.627
20	<i>250</i>	0.60	357	0.503	0.490
20	<i>500</i>	0.60	360	0.555	0.481
20	125	<i>0.60</i>	324	0.436	0.627
20	125	<i>0.75</i>	354	0.507	0.546
20	125	<i>1.00</i>	360	0.569	0.476

Table 2 shows that all three factors influence measures of the network. As each factor increases, the total links and average node degree increase correspondingly, but, the degree centrality decreases. Of the three factors, the population size has the most influence on these measures.

Egocentric Network The egocentric network is extracted from the declarative memory of each ACT-R agent. Figure 3 shows Agent-0s egocentric point-of-view. All relationships are weighted based on the agents activation values. Figure 4 shows a combined egocentric network of all the agents egocentric networks. We considered either a tie of $n_i \rightarrow n_j$ or $n_j \rightarrow n_i$ sufficient for inclusion in the combined network. We, thus, expect that the activation values found in each ego-net for ties $n_i \rightarrow n_j$ and $n_j \rightarrow n_i$ to differ. Nevertheless, when we compare Figure 2 and Figure 4, they have similar structures.

When we increase the activation threshold, we do see that links are eliminated as the minimal activation value neces-

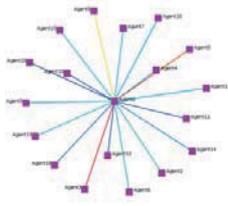


Figure 3: Egocentric network of an individual agent (name = Agent-0, population size = 20, running time = 125, configuration = Hallway)

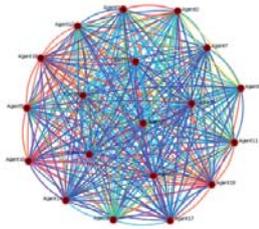


Figure 4: Egocentric network of all agents (population size = 20, running time = 125, configuration = Hallway)

sary for inclusion in the network increases, i.e. we see a layered network consisting of a strong set of relationships at its core and more casual ones at the periphery. Figure 5 shows the changes to the network when we increase the activation threshold. It appears that Figure 5 part c has a smaller but strong network because the memory retention between the existing nodes are relatively high.

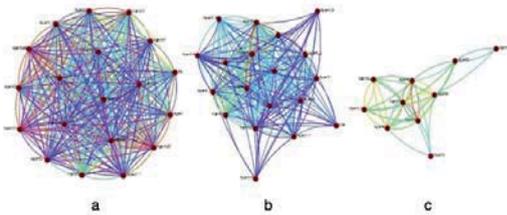


Figure 5: Egocentric networks with applied filters of a: -3.5 , b: 0.0 , c: 1.0

Example Pairwise Comparison

To examine the effect of cognitive constraints, we compared the activation levels of declarative memory chunks representing other agents in memory.

In a simple case (from a test run), Agent-0 remembers Agent-1 at about 0.2466, while Agent-1 remembers Agent-0 at about 0.5342. The higher the activation value, the better one agent remembers another, and here, we can see that Agent-0 does not remember Agent-1 as well as Agent-1 remembers Agent-0. This asymmetry can be due to Agent-0 not really paying attention to Agent-1, despite the simple fact that they had met.

In fact, looking at the ground truth from the system logs, Agent-0 and Agent-1 met each other 32 times in 125 seconds. Thus, despite meeting about once every four seconds, Agent-

0 did not remember Agent-1 as well as it might have. As a matter of fact, when a minimal memory threshold of 0.0 was applied (n.b. activation values can be negative), Agent-0 had no memory of Agent-1. A threshold analysis, which we discuss next, clearly shows the asymmetry of ties in memory (in this case, “A knows B but B does not know A”).

Based on this kind of asymmetric behavior, we suggest that the main impact of this kind of analysis on social network research will be in the realm of asymmetric relations. Ultimately, we expect to see that there is a distinction that needs to be made between knowledge of a relationship (among many possible relationships) and the attentional importance of that relationship.

Example Activation Cutoff

In Figure 5, we find another interesting story. When we set the threshold at 0.0, Agent-10 loses many of its connections between other agents. When the threshold is equal to -3.5 , Agent-10 has relations with all other agents (19 relations in total); but Agent-10 agent only has 9 relations left after the threshold was applied. After checking the log file, we found that the Agent-10 had multiple interactions (at least 13 times) with every agent. Most of these interactions, however, took place at the beginning of the simulation after this initial period Agent-10 was isolated at end of a hallway. The activation value between Agent-10 and the other agents continued to decay to values frequently below 0.0, with values ranging between -1.07 and 0.11. This case directly illustrates that not only does the frequency of interaction influence memory activation or the ties strength in memory but also the time and sequence of interactions.

Discussion and Conclusions

These results show how several common effects of cognition often influence network growth and shape. In this study, we created a multi-agent social network simulation that provides us a very flexible platform to examine factors that influence the development and maintenance of social networks in memory. Based on a review of the literature, we identified and modeled ecological (population size, map configuration, and run time) and cognitive factors, with the cognitive factors represented using memory activation parameters.

We conducted 27 runs of our simulation to test our model. The results indicate that all three factors influence the networks total links, average degree, and degree centrality. As each factor increases, the total links and average node degree increase correspondingly, but the as expected, the networks degree centrality decreases. Of these three factors, the population size has the most influence on the network measures. The effect of running time is not as significant as we expected, and shows plateauing after the 250s run. The large running time also weakens the effect of map configuration, because it provides agents enough time to travel around the whole map.

Taking advantage of the ACT-R memory mechanism, we were able to model the ego-nets of 1080 agents, and combine

those networks to compare the agents declarative representation of their friends network to the ideal network (or ground truth) generated from VIPERs log. We found the structure of ideal network and merged egocentric network to be similar (see 2 and 4). However, this similarity ends when we apply thresholds to the link weights in the merged egocentric networks (see Figure 5), where higher thresholds result in less connected networks that bear little resemblance to the ideal network. Semantically, this difference shows that memory limits how much of the ideal network an agent can remember well.

Future avenues of work will build upon some of the more interesting issues. First, we would do further analysis of normalized activation thresholds to see if these reliably effect either the network topology of the interaction network, or the topology of the agents declarative representation. Second, we would run more agents, because our test systems were kept deliberately small. Finally, we would analyze the effects of cognition on network measures analogous to Dunbar's Number.

Acknowledgments

DTRA (HDTRA1-09-1-0054) supported this work. Jaeyhon Paik provided technical support and Geoffery P. Morgan useful comments. CASOS developed ORA which proved invaluable as an analysis tool.

References

- Allen, T. (1977). Managing the flow of technology: Technology transfer and the dissemination of technological information with the R&D organization.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*, 1036–1060.
- Arcangeli, A., Eidus, I., & Wright, C. (2009). Increasing memory density by using KSM. In *Proceedings of the linux symposium* (p. 1928).
- Axelrod, R., & Hammond, R. (2003). The evolution of ethnocentric behavior. In *Midwest political science convention, chicago, il*.
- Carley, K. (1991). A theory of group stability. *American Sociological Review*, 331–354.
- Carley, K. (1992). Organizational learning and personnel turnover. *Organization Science*, 20–46.
- Carley, K., & Newell, A. (1994). The nature of the social agent. *Journal of mathematical sociology*, *19*(4), 221–262.
- Dunbar, R. (1998). *Grooming, gossip, and the evolution of language*. Harvard Univ Pr.
- Gonzalez, C., Lerch, J., & Lebiere, C. (2003). Instance-based learning in dynamic decision making. *Cognitive Science*, *27*(4), 591–635.
- Jovina, I., Lebiere, C., Martin, J., & Gonzalez, C. (2011). Intergroup prisoners dilemma with intragroup power dynamics. *Games*, *2*(1), 21–51.
- Kraut, R., Fussell, S., Brennan, S., & Siegel, J. (2002). Understanding effects of proximity on collaboration: Implications for technologies to support remote collaborative work. *Distributed work*, 137–162.
- Lebiere, C., Gonzalez, C., Dutt, V., & Warwick, W. (2009). Predicting cognitive performance in openended dynamic tasks a modeling comparison challenge. In *Proceedings of the 9th international conference on cognitive modeling, manchester, uk*.
- McCarty, C., Killworth, P., Bernard, H., Johnsen, E., & Shelley, G. (2001). Comparing two methods for estimating network size. *Human Organization*, *60*(1), 28–39.
- Morgan, J., Morgan, G., & Ritter, F. (2010). A preliminary model of participation for small groups. *Computational & Mathematical Organization Theory*, 1–25.
- Prietula, M., & Carley, K. (2001). Boundedly rational and emotional agents: Cooperation, trust and rumor. *Trust and Deception in Virtual Societies*. Edited by Cristiano Castelfranchi and Yao-Hua Tan. Kluwer Academic Publishers, 169–194.
- Reitter, D., & Lebiere, C. (2010). A cognitive model of spatial path-planning. *Computational & Mathematical Organization Theory*, 1–26.
- Simon, H. A. (1984). *Models of bounded rationality: Economic analysis and public policy*. The MIT Press.
- Zhao, C., Kaulakis, R., Morgan, J. H., Hiam, J. W., & Ritter, F. E. (2012). Modeling a cognitively limited network in an agent-based simulation. In *Proceedings of the 34th conference of the cognitive science society*. Sapporo, Japan. (SUBMITTED)
- Zhao, C., Kaulakis, R., Morgan, J. H., Hiam, J. W., Sanford, J. P., Ritter, F. E., et al. (2012). Socio-cognitive networks: Modeling the effects of space and memory on generative social structures. In *Proceedings of the 21st conference on behavior representation in modeling and simulation*. Amelia Island, FL. (ACCEPTED)

Emotion Detection by Event Evaluation using Fuzzy Sets as Appraisal Variables

Michael Kotzyba, Barbara Deml, Hendrik Neumann (michael.kotzyba@ovgu.de)
Otto-von-Guericke University, FMB IAF-Human Factors Engineering, 39106 Magdeburg, Germany

Stefan Gluege, Kim Hartmann, Ingo Siegert, Andreas Wendemuth
Otto-von-Guericke University, FEIT IESK-Cognitive Systems, 39106 Magdeburg, Germany

Harald C. Traue, Steffen Walter
University of Ulm, Medical Psychology, 89075 Ulm, Germany

Keywords: cognitive modeling; emotion detection; appraisal theory

Introduction

A very important and challenging task in cognitive science is the detection and modeling of human emotions. On the one hand, computers could benefit from that, because emotions play a significant role in rational decision making, perception, learning, and a variety of other important cognitive functions. On the other hand there is a need for genuinely intelligent computers that adapt and interact with human users in a natural way. To achieve this goal, computers need the ability to recognize and to express emotions (Picard, 1997).

Even for humans it is difficult to recognize the emotions of other persons. But this is not always evident because we use knowledge about the situational circumstances. For example, if a person is getting failure messages for a long time while using a computer program, it will be obvious that the facial expression is anger. Without this situational knowledge it is much more difficult to be empathetic. Therefore, computers should also use knowledge about the situation or the event to detect human emotions.

The appraisal theory is an emotion theory providing a model for evaluating a situation or an event in terms of relevant variables. In this work we present a framework which maps the evaluation process from appraisal theory to a fuzzy model in order to derive the emotions of a user in a specific situation/event.

At first we give a short introduction to appraisal theory. Then we introduce our fuzzy model of appraisal theory. Finally the model is evaluated on recorded data of a human-computer interaction in a Wizard-of-Oz scenario.

Appraisal Theory of Emotion

Almost all emotion theories assume that the specific kind of emotion experienced depends on the result of an evaluation process of relevant events (Scherer, Schorr & Johnstone, 2001). Thereby it is evaluated how these events affect the well-being of the organism. Appraisal theories attempt to specify the nature of criteria used in evaluation in terms of different appraisal variables or dimensions. Examples for these dimensions are “goal significance” representing the goals and needs that are of high priority at the moment (e.g.

the goal of survival, maintaining social relationships or win a game), or “urgency” representing the need for an action.

In Ellsworth and Scherer (2003) it is suggested, that the given appraisal variables allow to deduce an emotion as the most probable emotional reaction to a certain event. For example, joy or happiness will occur, if the appraisal values for the variable “goal significance” are high, while the value for “urgency” is low etc.

Implementation of Appraisal Theory

In this section it is described how the model has been developed and how it works using a fuzzy model (Kahlert & Frank, 1993; Michels et al., 2006)

Appraisal Variables as Fuzzy Sets

In this implementation ten different appraisal variables are used (cf. table 29.2 in Ellsworth & Scherer, 2003). To operationalize the appraisal variables and their linguistic values, each appraisal variable is modeled as a fuzzy set. The fuzzy sets consist of several fuzzy variables, depending on the number of postulated values of the appraisal variable. For example, “urgency” can have five different values (“very low”, “low”, “medium”, “high”, “very high”) and thus it is modeled by five corresponding fuzzy variables. Each appraisal variable is mapped to a uniform number range from 0 to 100. The corresponding fuzzy variables are distributed uniformly in this range. For simplicity we use a triangle function to model a fuzzy variable.

Using all ten fuzzy sets, we are able to model both the emotional state of the user and a possible evaluation of an arising event in terms of appraisal variables. In Figure 1 a systematic overview of the framework is given with an event and the user state with ten appraisal variables (av).

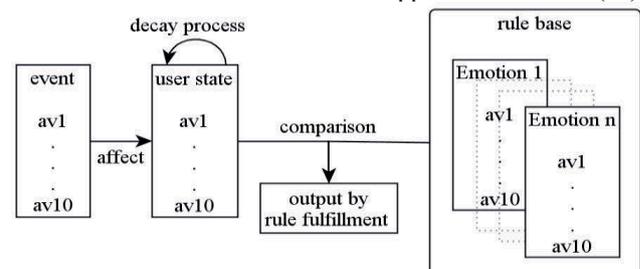


Figure 1: Conceptual view of the framework

Adapting the Emotional State

According to appraisal theory, the emergence of an event triggers an evaluation process. To model this process, the fuzzified appraisal variables of an event can be used to adapt the corresponding variables in the user state. For example, an event with high “goal significance” should affect the user’s condition for “goal significance” positively. In which manner the events affect the user’s state is not trivial because humans interpret events and their relevance differently. In the simplest way, the appraisal variables of an event can affect the user’s emotional state directly by overwriting it (cf. left hand side of Fig. 1).

Derivation of Emotions

As stated above, the appraisal variables can be used to deduce an emotion as the most probable emotional reaction to a certain event. Each emotion is described by appraisal variables with specific values. From this specification one can generate a rule base for each emotion. To derive an emotion, the postulated emotion profile and the user state can be compared to calculate the fulfillment of the emotion rule base (cf. Fig. 1).

Experimental Evaluation

To test our framework, we use a dataset of 18 test persons playing the popular game “concentration” aka “memory” on a computer. While playing, the test person is supported by a computer assistant, imitated by the investigator (Wizard-of-Oz experiment). Each test person plays six games, also called experimental sequences (abbr.: ES). The ES are designed to provoke different emotional states. We focus on the ES2 and ES5, because they represent emotional extrema: In ES2 a simple card set, low time constraint and positive feedback by the assistant provoke a pleasant or positive feeling i.e. happiness. ES5 includes a difficult card set and the assistant gives negative feedback to provoke negative feelings, i.e. anger. For more details concerning the dataset, see (Walter et al., 2011).

To apply the model, all possible events (like a hit or neg. feedback) were extracted and possible evaluations regarding the corresponding appraisal variables were formulated. For example, it is plausible that turning over a pair of matching cards is considered as goal significant while receiving neg. feedback is considered as unpleasant. For simplicity it is assumed that an event affects the emotional state of the user

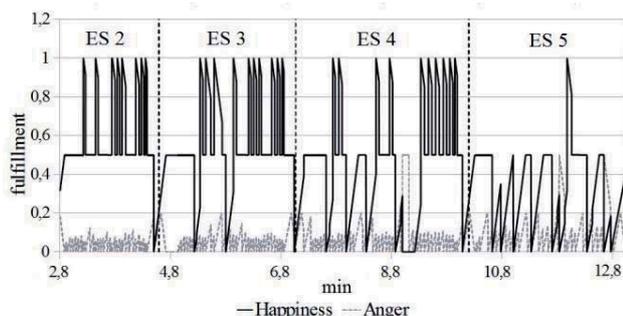


Figure 2: Rule fulfillment over ES2 up to ES5

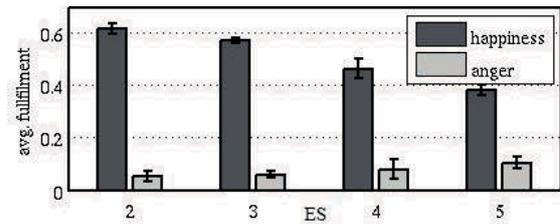


Figure 3: Average rule fulfillment over all test persons

directly. Since emotions have a short duration, a decay process is implemented (cf. “user state” in Fig. 1). At this point it is assumed that a “neutral” user state will be reached if all appraisal variables receive the value “medium”.

In figure 2 the results for the ES2 up to ES5 of one test person are illustrated. In this experiment we only use the rule base to derive happiness and anger. The values in the diagram represent the fulfillment of the two emotion rules, calculated by the model, based on the events that occurred in the sessions. In comparison to ES5 the values in ES2 for happiness are higher and anger occurs only slightly. In ES5 the values for happiness are reduced and anger is increased. Figure 3 shows the average values of rule fulfillment for each ES over all test persons. Here one can discover the same pattern for emotions as in Fig 2. These results confirm that the appraisal theory is applicable in terms of the presented model.

Outlook

Humans interpret events and their relevance differently. For that reason the focus of future work lies in event evaluations via appraisal variables and whose derivation from available data. Possible sources can be audio, video and psychobiological data. Further the dialog between the user and the application, as well as the environment and the personality or the user’s mood should be considered, too.

Other important issues are the decay process of emotions and the occurrence of multiple events.

References

- Ellsworth, P. C., & Scherer, K. R. (2003). Appraisal Processes in Emotion. Handbook of affective sciences. Oxford University Press. New York & Oxford.
- Picard, R. W. (1997). Affective Computing. MIT Press.
- Kahlert, J., & Frank, H. (1993). Fuzzy-Logik und Fuzzy-Controll. Vieweg Verlag
- Michels, K., Klawonn, F., Kruse, R., & Nürnberger, A. (2006). Fuzzy Control: Fundamentals, Stability and Design of Fuzzy Controllers. Springer
- Scherer, K. R., Schorr, A., & Johnstone, T. (2001). Appraisal Processes in Emotion: Theory, Methods, Research. Oxford University Press. New York & Oxford.
- Walter, S., Scherer, S., Schels, M., Glodek, M., Hrabal, D., Schmidt, M., Böck, R., Limbrecht, K., Traue, H. & Schwenker, F. (2011). Multimodal Emotion Classification in Naturalistic User Behavior. In Proceedings of the 14th HCI'11. LNCS 6763, 603 - 611

Modeling behavior and performance of air traffic controllers using coloured petri nets

Hardy Smieszek (hardy.smieszek@zmms.tu-berlin.de)
Technische Universität Berlin, Centre of Human-Machine Systems,
10587 Berlin, Germany

Abstract

Airport infrastructure is often unable to grow at the same rate as the steadily growing volumes of air traffic and ground controllers face increasing workloads, because they have to deal with more aircraft in the same time. This can be counteracted by making changes to the system or system conditions, e.g. by implementing computerized supporting systems. Simulations are needed to investigate the impact of changing system conditions on the human operator. In addition to modeling the airport traffic control system in general, it is also necessary to model the cognitive processes and behavior of the ground controller. Based on basic guidelines for the development of cognitive simulation, an approach for the development of a coloured Petri net model of aerodrome air traffic control is presented.

Keywords: Task Analysis, Aerodrome Air Traffic Control, Coloured Petri Nets.

Introduction

Facing steadily growing volumes of air traffic, ground controllers encounter increasing efficiency demands and excessive workloads as they have to deal with more aircraft in the same time. At the moment the aviation industry is one of the safest modes of transport. Following Hollnagel, Woods and Leveson (2006) it is the inherent resilience of the system that makes it safe. In aviation it is the variability of human performance which enables air traffic controllers (ATCOs) and pilots most of the time to act in a safe way and to take the right action at the right time (Stroeve, Everdij & Blom, 2011). Above all ATCOs are responsible for the safe and efficient handling of air traffic where they regularly have to reach a tradeoff between efficiency and thoroughness if they are to be successful (Hollnagel, 2009). With increasing air traffic this tradeoff has to be shifted towards efficiency, which can lead to erroneous actions and increasing numbers of incidents and accidents. This effect can further be influenced by so called performance conditions, which are by definition a set of environmental, personal and systemic variables which can alter the possibility of erroneous actions (Center for Chemical Process Safety, 1994). To help ATCOs conduct their tasks safely and efficiently, changes to the system or system conditions will be introduced, e.g. by implementing computerized supporting systems like A-SMGCS (Advanced Surface Movement Guidance and Control System; EATMP, 2005). For the investigation of the impact

of changing system conditions on the human operator a model of the human behavior and performance is needed.

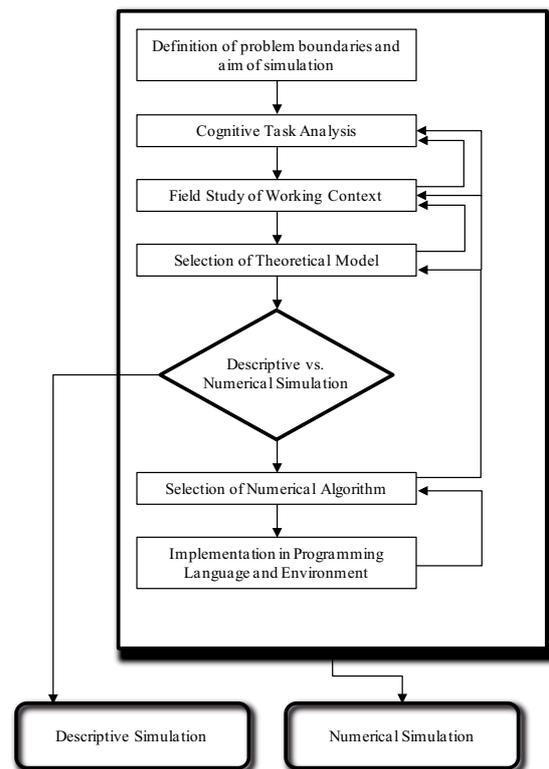


Figure 1: Basic guidelines for the development of cognitive simulation (Cacciabue, 1998).

Development of cognitive simulation

Following the basic guidelines for the development of cognitive simulation given by Cacciabue (1998) shown in Figure 1, the first step is the definition of the problem boundaries and aim of the simulation. For modelling purposes it is not only necessary to model the airport traffic control system in general, it is also necessary to model the cognitive processes and behavior of the ground controller. An essential requirement for this is exact knowledge about the tasks of ground controllers.

Cognitive task analysis and field study of working context

The second and third steps are a (cognitive) task analysis and a field study of the working context. Existing task analyses were reanalyzed in terms of their focus and

methods for several sites across Europe, including analyses by EUROCONTROL (Buck, Biemans, Hilburn & Van Woerkom, 1996; Tavanti & Bourgois, 2006), German Air Navigation Services (DFS; Human-Factors-Consult, 2009) and Royal Air Force Institute of Aviation Medicine (Cox, 1994). Through these analyses, a widespread illustration of the tasks of the ground controllers has been generated and a model of the action phases of the ground controller based on action regulation theory has already been introduced elsewhere (Smieszek, Huber & Jürgensohn, 2011). Several main tasks and sub-tasks are described in further detail and a prototypical action sequence is created which will form the basis for the model construction. The review of task analysis will be validated with the help of expert interviews and field studies in two airport control towers of the Berlin airports.

Theoretical Model

The fourth step, as proposed by Cacciabue (1998), is the selection of a theoretical model, which in this case will be the contextual control model (COCOM) developed by Hollnagel (1993). Within this framework it is assumed that all human behavior is essentially influenced and the choice of the next action is determined by the actual context as it is also proposed by the situated cognition approach (e.g. Brown, Collins & Duguid, 1989).

Selection of numerical algorithms and implementation in programming language

To gain a numerical simulation, the final two steps are the selection of numerical algorithms and the implementation in programming language. For this purpose the framework of Coloured Petri Nets is chosen. Coloured Petri Nets is a language for the modeling and validation of concurrent and distributed systems and other systems in which concurrency plays a major role (Jensen & Kirstensen, 2009). It provides both a graphical representation and a mathematical description of the modelled system. Firstly a Coloured Petri Net will be constructed which models the normal process of aerodrome air traffic control. Afterwards Fuzzy Logic (Zadeh, 1965) terms will be introduced to include the influence of several performance conditions on the operation of the system. With the help of the Petri Net model the investigation and evaluation of the impact of system changes on human behavior and performance will be possible.

Conclusion

Due to the dramatic changes in the air traffic control sector especially the growing traffic levels it is necessary to study the effects of changing system conditions on cognitive processes and the behavior of aerodrome air traffic controllers. This can be done by using models and simulations. An approach has been presented which will provide a model and simulation both of the airport control system and cognitive processes and the behaviour of the tower controller based on Coloured Petri Nets. The

theoretical framework is provided by the contextual control model (Hollnagel, 1993). This describes how the context influences the behavior of the individual. This framework will be the basis for the Coloured Petri Net model which will provide the insight to the question how controllers will behave under changing system conditions.

References

- Brown, J. S.; Collins, A. & Duguid, S. (1989). Situated cognition and the culture of learning. *Educational Researcher*, 18 (1): 32–42.
- Buck, S., Biemans, M. C. H., Hilburn, B. G., & Van Woerkom, P. T. L. M. (1996). *Synthesis of functions (NLR Report TR-970545L)*. Amsterdam: NLR.
- Cacciabue, P.C. (1998). *Modelling and Simulation of human behavior in system control*. London: Springer
- Center for Chemical Process Safety (1994). *Guidelines for Preventing Human Error in Process Safety*. New York: American Institute of Chemical Engineers.
- Cox, M. (1994). *Task Analysis of selected operating Positions within UK Air Traffic Control Volume 1: Main Report & Volume 2: Appendices*. Farnborough, Hampshire: Royal Air Force Institute of Aviation Medicine.
- EATMP. (2005). *Definition of A-SMGCS Implementation Levels*. EUROCONTROL.
- Hollnagel, E. (1993). *Human Reliability Analysis: Context and Control*. London: Academic Press.
- Hollnagel, E. (2009). *The ETTO Principle: Efficiency-Thoroughness Trade-Off – Why things that go right sometimes go wrong*. Farnham: Ashgate.
- Hollnagel, E., Woods, D.D., & Leveson, N. (2006). *Resilience Engineering – Concepts & Percepts*. Aldershot. Ashgate.
- Human-Factors-Consult, H. F. C. (2009). Bericht API – Aufgabenanalyse.
- Jensen, K. & Kirstensen, L.M. (2009). *Coloured Petri Nets – Modelling and Validation of Concurrent Systems*. Berlin: Springer
- Smieszek, H., Huber, S. & Jürgensohn, T. (2011). Ein Integratives Prozessmodell der Hauptaufgaben von Rollverkehrsleitern als Grundlage für Modellierung und Simulation. In Schmid, S., Adenauer, J., Elepfandt, M., Lichtenstein, A. (eds.). *Tagungsband der 9. Berliner Werkstatt Mensch-Maschine-Systeme: Reflexionen und Visionen der Mensch-Maschine-Interaktion*, Berlin, Düsseldorf: VDI Verlag.
- Stroeve, S.H., Everdij, M.H.C., Blom, H.A.P. (2011). *Studying hazards for resilience modeling in ATM – Mathematical Approach towards Resilience Engineering in ATM (MAREA). First SESAR Innovation Days: Ecole Nationale de l'Aviation Civile (ENAC), Toulouse, France*.
- Tavanti, M., & Bourgois, M. (2006). *Control Tower Operations: A Literature Review of Task Analysis Studies*. Rome, Italy: EUROCONTROL.
- Zadeh, L.A. (1965). Fuzzy Sets. *Information and Control*, 8, 338-353

A Framework for Task Accomplishment Using an ACT-R Simulation

Michael Beckmann, Ulaş Yılmaz, Gloria Pöhler & Anne Wegerich
({mbe, uyi, gpo, awe}@mms.tu-berlin.de)

Chair of Human-Machine Systems, Department of Psychology and Ergonomics,
Technische Universität Berlin, Franklinstr. 28/29, 10587 Berlin

Keywords: Machine Learning; Cognitive Modeling; Process Simulation; Error Recognition; Error Prevention

Motivation

Modern business concepts, like *Industrial Product-Service Systems* (IPS²) (Meier, Roy, & Seliger, 2010), pose greater demands on human operators for managing and maintaining the involved technical systems. A missing assistance application to counteract this change leads to an increase in operator errors and a decrease of the overall robustness of the socio-technical system.

Cognitive architectures, like ACT-R (Anderson et al., 2004), provide the possibility of a human centered and perfect modeling of task accomplishments. This allows the prediction of steps in a process with high risks for human mistakes, as well as the recognition of human mistakes. Furthermore motion capture systems (Bregler, 2007) and advances in the field of machine learning and action recognition can provide additional (real-time) information about human interaction. To ensure an optimal production cycle the factory machinery is equipped with additional sensors. The sensory information is used to predict machine failure to schedule maintenance task in advance, before the machine breaks down.

Thus, real-time simulation of a cognitive model with access to the state space of the technical system and knowledge about human action can predict risks within a task and is able to recognize invalid execution paths. The description of the observed mistake or of a high risk situation allows an increase of the robustness of the overall system, which is the goal of subproject B5 of the Collaborative Research Center Transregio 29.

In this article we introduce a simple and effective realization of a more complicated interactive warning framework, in order to avoid operator mistakes during maintenance.

Framework

To optimize the robustness of socio-technical systems the framework consists of a module for the cognitive model (ACT-R), which enables real-time simulation of the accomplishment of a task with respect to human parameters. This module receives information about the state space of the involved technical systems from a technical module. Information about human actions is provided by a gesture module. The cognitive module evaluates the information of the other data sources to determine valid execution paths. If a deviation from expected human actions is observed and for each a report is generated.

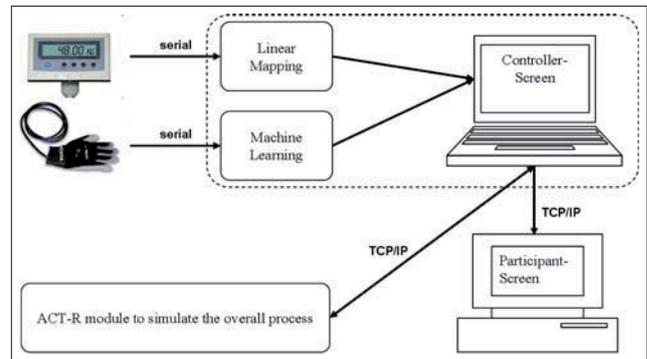


Figure 1: Overall system setup

This article describes an initial version of the framework. The modeled task is to reach a given target weight by putting several interaction objects with different shapes and weights on an electronic scale. This task is based on the experiment by Lovett, John, and Anderson (1996).

Data and Input

The three input sources of the test setup are illustrated in Figure 1: a scale, a data glove and an expert to validate the gestures of the participants in a Wizard-of-Oz manner. The scale is a *PCE-TS* platform scale that has an RS-232-interface and a capacity of 60kg x 5g (PCE Instruments UK Ltd., 2011). The current object put on the scale is recognized by a linear mapping procedure using the known real weights of the objects. The real weight of the object is then mapped to a virtual weight the participant sees on his screen. The objects under consideration and the appropriate gesture for holding these objects are shown in Figure 2.



Figure 2: Objects, their weights and the corresponding hand gestures while holding each one of them

Hand movements are recorded using an *X-IST Wireless DataGlove* that is equipped with tilt and bend sensors (*X-IST Data Glove*, 2008): two sensors on the thumb and three on each of the remaining fingers. Hand gestures are then detected by measuring the relative bend of these sensors at the finger joints. The gesture recognition module normalizes the data from the data glove to account for inter-individual differences. Afterwards the module classifies the data using a supervised machine learning algorithm into one of the three hand gestures each of which corresponds to the grabbing of a weight (See Figure 2).

Simulation

The ACT-R simulation enables the identification of operator mistakes during a task. It incorporates state space information of the technical system and gestures by the subject for rule selection. The simulation displays a warning only when a subject executes an invalid gesture during the experiment in order to be less intrusive.

The simulation considers all non-circular valid execution paths for solving the task up to the next expected gesture. All these paths are in a conflict set until the gesture module resolves the conflict by providing information about the next gesture performed by the subject. When the observed gesture is in the conflict set the simulation continues, otherwise an error report is generated and the simulation waits for a correction by the subject. The incorporation of preparatory gestures enables the recognition of an invalid action before it impacts the system state. In this case the preparatory gesture is the grasping of a weight and the system state equals the weight on the scale.

In our setup, the overall task is subdivided into three phases, as illustrated in Figure 3. The first phase deals with the identification of the weight of each interaction object. In this stage, both the human and the model try to identify the weight of each interaction object. In the second phase, subjects are asked to select one out of two strategies to reach the given target weight. In the “overshoot” strategy, the target weight can be achieved by starting with a bigger weight and subtracting all following weights from it. The second strategy is referred to as “undershoot” strategy, in which the subject starts with a weight smaller than the desired one and add all following interaction objects, so that the sum of them matches the given target weight. In the third phase, the subjects place the remaining interaction objects on the scale according to their chosen strategy to reach the target weight. An improper action in the sense of this simulation is the selection of a strategy or weight which does not lead to the solution of the task.

Conclusions and Future Work

In this article we described an interactive warning framework. With this framework operator mistakes are detected online and task accomplishment is ensured. Participants are asked to reach a given target weight on an electronic scale using different interaction objects. Gestures of the participants are

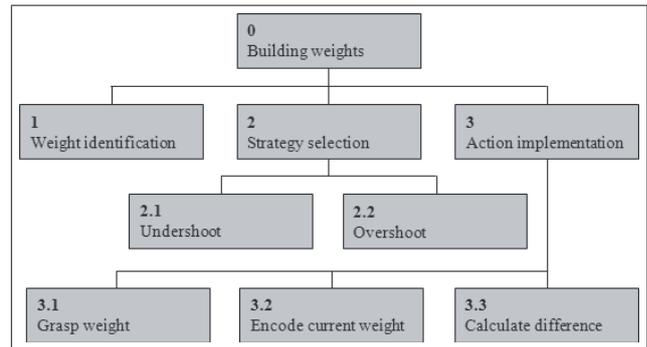


Figure 3: ACT-R tree diagram of the human events

detected using a data glove. These actions are validated by an ACT-R simulation that raises warnings in case of invalid execution paths.

A next step in the development of the framework is to test more complex technical systems and user interactions found in *IPS²* scenarios. The consideration of all valid execution paths is a deviation from the human decision making and poses a problem when the simulation should consider errors due to memory degradation. In this case all valid execution paths can be calculated by cloning the simulation when several rules can be selected which do not contain observable actions. The conflict resolution set is constructed from all concurrent simulations. Simulations which are not consistent with observed user behavior are discarded. An incorporation of the machine learning module into an ACT-R motor module enables the simulation to incorporate execution times for various actions. This allows the simulation to generate reports when no activity is observed, which can be interpreted as uncertainty of the operator.

References

- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004, October). An integrated theory of the mind. *Psychol Rev*, *111*(4), 1036–1060.
- PCE Instruments UK Ltd. (2011). *Platform balance pce-ts series*. Retrieved February 21, 2012, from <http://www.industrial-needs.com/technical-data-scales/platform-balance-pce-ts.htm>
- Bregler, C. (2007, nov.). Motion capture technology for entertainment [in the spotlight]. *Signal Processing Magazine, IEEE*, *24*(6), 160 -158.
- Lovett, M. C., John, & Anderson, R. (1996). History of success and current context in problem solving: Combined influences on operator selection. *Cognitive Psychology*, *31*, 168–217.
- Meier, H., Roy, R., & Seliger, G. (2010). Industrial product-service systems - *ips²*. *CIRP Annals - Manufacturing Technology*, *59*(2), 607 - 627.
- X-IST Data Glove. (2008). *X-ist data glove - virtual reality glove*. Retrieved February 21, 2012, from <http://www.vrealities.com/x-ist.html>

Performance Modelling of Interface Dynamism in Motor Skills Acquisition

Olurotimi R. Akinlofa (o.r.akinlofa@rgu.ac.uk)

Patrik O'Brian Holt (p.holt@rgu.ac.uk)

Eyad Elyan (e.elyan@rgu.ac.uk)

IDEAS Research Institute, Robert Gordon University
St Andrew Street, Aberdeen, AB25 1HG, UK

Keywords: Skill Acquisition, Learning, Interface Design.

Introduction

Instructions are central to many human skill acquisition processes e.g. like those used for pilot training (Dennis & Harry, 1998). Recent advances in computing technologies have expanded the scope of computer based instructional delivery especially where safety and cost may preclude the use of traditional training systems. The effectiveness of such CBT systems has been subjected to considerable research (Höffler & Leutner, 2007). An important aspect of research into the effectiveness of such instructional methods, which is relevant to the study reported here, is the benefit of dynamic over static components of instructional interfaces used in the acquisition of procedural motor skills as typical in aviation engineering training simulators.

Akinlofa, Holt, and Elyan, (under revision), propose a model (Figure. 1) to explain the observed benefit of dynamic visualisations compared with statics for learning novel procedural motor skills by aviation engineering trainees. Following on, a representative sub-step from the study is modelled using the ACT-R 6.0 architecture to examine the post-learning task performances of the different learner groups.

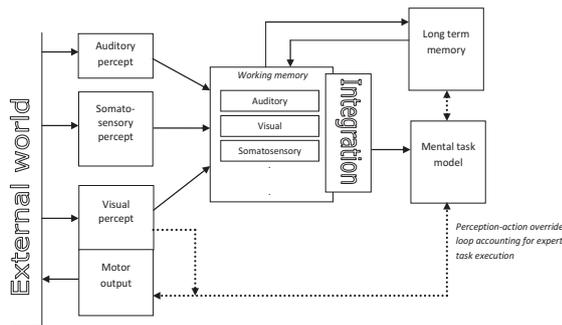


Figure 1: Information-processing model for learning a procedural motor task

Cognitive Modelling

It is assumed that different declarative knowledge structures, dependent on the instructional formats, are created for the sequence of component spatial states in a rotation movement. Static diagrammatic instructions can only afford the initial and final states of the rotated component while dynamic, video instructions will provide

knowledge of the start and end states as well as all transitory states in between. Subsequent motor performance is driven by a sequential retrieval of the states, interspersed with a random strategy if retrieval fails. Figure 2 shows that the representative model for the static condition is constrained to a random strategy while the dynamic model utilises a mixed strategy.

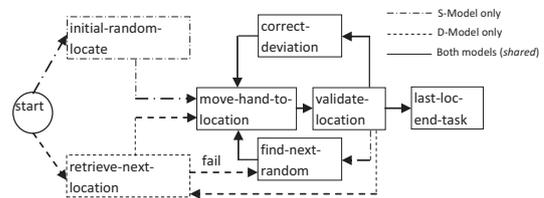


Figure 2: Schematic outline of model's productions

Motor performance for the models is implemented as a chained sequence of unit movement vectors simulating the transition of a selected reference point of the rotated component in 2-D space. The number of unit movement vectors in the movement sequence as well as their individual directions is stochastically dependent on the current position in the trajectory and the selected productions firing per cycle of cognitive processing.

The default mechanism of the motor module of ACT-R 6.0 was not suitable for our model design as it utilises Fitts' law to calculate movement execution time towards a specified target. Additionally, it calculates incremental positions along a movement path for specified start and end positions only. Our model's movement strategy however specifies only a start position, while the end position is stochastically determined by a fixed magnitude, variable direction, unit movement vector. Furthermore, as movement is implemented by a sequence of unit vectors, the transitory point from vector to vector must be modelled accurately to ensure uniform and continuous acceleration throughout the trajectory. Therefore, the default motor module is redefined through an adaptation of the dynamic cost optimisation approach for the mathematical modelling of human hand movements (Flash & Hogan, 1985). By using the minimisation of the time integral of the square of jerk on the curved component rotation trajectory, point-to-point movement is represented by the insertion of intermediate points (at times t_1, t_2, \dots, t_n) between the start and end positions. The entire trajectory is then modelled through a shifting boundary condition method across the range of via

points bounded by $t=0$ and $t=t_f$ according to the following equations:

for all times $t \leq t_n$

$$x^-(\tau) = \frac{t_f^5}{720} (\mu_x (\tau_n^4 (15\tau^4 - 30\tau^3) + \tau_n^3 (80\tau^3 - 30\tau^4) - 60\tau^3 \tau_n^2 + 30\tau^4 \tau_n - 6\tau^5) + c_x (15\tau^4 - 10\tau^3 - 6\tau^5)) + x_0$$

$$y^-(\tau) = \frac{t_f^5}{720} (\mu_y (\tau_n^4 (15\tau^4 - 30\tau^3) + \tau_n^3 (80\tau^3 - 30\tau^4) - 60\tau^3 \tau_n^2 + 30\tau^4 \tau_n - 6\tau^5) + c_y (15\tau^4 - 10\tau^3 - 6\tau^5)) + y_0$$

and for all times $t \geq t_n$

$$x^+(\tau) = \frac{t_f^5}{720} (\mu_x (\tau_n^4 (15\tau^4 - 30\tau^3 + 30\tau - 15) + \tau_n^3 (-30\tau^4 + 80\tau^3 - 60\tau^2 + 10)) + c_y (-6\tau^5 + 15\tau^4 - 10\tau^3 + 1)) + x_f$$

$$= x^-(\tau) + \frac{\mu_x t_f^5 (\tau - \tau_n)^5}{120}$$

$$y^+(\tau) = \frac{t_f^5}{720} (\mu_y (\tau_n^4 (15\tau^4 - 30\tau^3 + 30\tau - 15) + \tau_n^3 (-30\tau^4 + 80\tau^3 - 60\tau^2 + 10)) + c_y (-6\tau^5 + 15\tau^4 - 10\tau^3 + 1)) + y_f$$

$$= y^-(\tau) + \frac{\mu_y t_f^5 (\tau - \tau_n)^5}{120}$$

where $\tau = t/t_f$; $\tau_n = t_n/t_f$; t_n is a via-point; μ_x, μ_y, c_x , and c_y are constants.

This affords accurate and continuous implementation of hand acceleration through the transition points in the movement vector sequence. The partial matching mechanism of the retrieval module is further utilised to simulate the inaccuracy of recalling component intermediate positions along the trajectory of rotation. As the model movement is implemented in 2-D Cartesian space, a sim-hook function is used to define matching inaccuracies on the x -coordinates. Additionally, an extension of the activation equation is used to define matching on the y -coordinate and a summation of the matching functions outputs is computed as the overall match score of a specific location in the movement space. This design, as depicted in Figure 3, is very flexible and could be a starting point for extending to 3-D spatial movement.

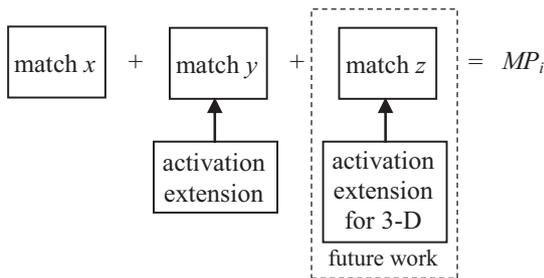


Figure 3: Spatial location partial matching

Results and Further Work

Comparison with human data shows that the model's quantitative predictions were accurate on latencies ($R^2=.98$, $RMSE=.52$) and trajectory tracking. This result supports an advantage of dynamism in the instructional interface for procedural skill acquisition. It further supports our hypothesis on the hybrid role of cognition in procedural

motor performance and validates the assumed strategies of task execution of initially attempting to recall a mental task image and resorting to a controlled stochastic method in the event of a retrieval failure. However, there are other factors, such as the cognitive abilities of the learners, which were controlled in the larger empirical study in contrast with the implementation of the computational models. Our result is further limited because the models reflect a single step of a procedural movement and implements the movement in 2-D space only. Further collaborative work will extend the model to cover the entire movement sequence of the experimental task reported in Watson, Butterfield, Curran, and Craig (2010). Our future work will also explore the execution of the task with 3-D spatial movement using the approach of extending the activation equation for spatial matching as outlined above.

Acknowledgments

This study was supported by the Petroleum Technology Development Fund (PTDF), Nigeria Grant: PTDF/TR/PhD/AOR/206/09.

References

- Akinlofa, O.R., Holt, P.O., & Elyan, E. (under revision). *Effect of interface dynamism on learning procedural motor skills*.
- Dennis, K.A., & Harry, D. (1998). Computer-Based Simulation as an adjunct to ab-initio flight training. *The International Journal of Aviation Psychology*, 8, 261–276.
- Flash, T., & Hogan, N. (1985). The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of Neuroscience* 5, 1688–1703.
- Höfler, T., & Leutner, D. (2007). Instructional animation versus static pictures: A meta-analysis. *Learning and Instruction* 17, 722–738.
- Watson, G., Butterfield, J., Curran, R., & Craig, C. (2010). Do dynamic work instructions provide an advantage over static instructions in a small scale assembly task?. *Learning and Instruction* 20, 84-93.

Explorations in ACT-R Based Language Analysis – Memory Chunk Activation, Retrieval and Verification without Inhibition

Jerry T. Ball (Jerry.Ball@wpafb.af.mil)

Air Force Research Laboratory, 711th Human Performance Wing
2620 Q Street, Bldg 852, WPAFB, OH 45433

Abstract

This paper explores the benefits and challenges of using the ACT-R cognitive architecture in the development of a large-scale, functional, cognitively motivated language analysis model. The paper focuses on ACT-R's declarative memory retrieval mechanism, proposing extensions to support verification of retrieved chunks, multi-level activation spread and carry over activation. The paper argues against the need for inhibition between competing chunks which is necessarily task specific.

Keywords: ACT-R; activation; retrieval; verification; inhibition; language analysis.

Introduction¹

Our team has been working on the research and development of a language analysis model (Ball, 2011a; Ball, Heiberg & Silber, 2007) within the ACT-R cognitive architecture (Anderson, 2007) since 2002 (Ball, 2003). Currently, the model comprises ~950 productions and over 58,000 declarative memory (DM) chunks. The model is capable of processing a broad range of English language constructions (www.doublertheory.com/comp-grammer/comp-grammar.htm; Ball, Heiberg & Silber, 2007) and is a component of a larger synthetic teammate model (Ball et al., 2010). The model accepts written input from single words to entire documents, and processes the input incrementally, one word or multi-word unit at a time. On a 64-bit quad-core Windows machine with 8 Gig RAM, the model incrementally processes ~130 words per minute (wpm) with the full 58,000 chunk mental lexicon and ~320 wpm with a smaller 22,000 chunk mental lexicon. The model processes ~145 wpm in ACT-R cognitive processing time which compares to adult reading rates ranging from 200-300 wpm. We are working on ways to improve the analysis rate of the model—which does not entail full comprehension—to bring it into closer alignment with adult reading rates (Freiman & Ball, 2010).

Our focus is on research and development of a general-purpose, large-scale, functional model that adheres to well established cognitive constraints on human language processing (HLP) (Ball et al., 2010). Two important constraints that we adhere to are *incremental* and *interactive* processing (Just & Carpenter, 1987; Altmann & Steedman, 1988; Tanenhaus et al., 1995; Gibson & Pearlmuter, 1998).

Adherence to these constraints precludes the use of computational techniques like algorithmic backtracking and staged analysis (i.e., independent tokenizing, part of speech tagging, syntactic analysis, semantic analysis, and pragmatic analysis) and limits the use of techniques like lookahead, underspecification and parallel propagation of constructed alternatives—all of which are mainstays of many computational linguistic systems.

ACT-R incorporates two architectural constraints, realized as serial bottlenecks, which largely determine incremental processing: 1) a single production can execute at a time, and 2) a single DM chunk can be retrieved at a time. In addition to these serial constraints which are the basis of *incremental* processing, ACT-R provides architectural support for parallel processing in the form of a parallel production selection mechanism based on utility, and a parallel DM retrieval mechanism based on activation. These parallel mechanisms are probabilistic and context dependent. The parallel/probabilistic/context dependent mechanisms provide the basis for *interactive* processing. They guide the processing of the language analysis model in directions that are likely to lead to a successful analysis given the current context and current input. The highly parallel retrieval mechanism is capable of selecting from existing DM chunks, but does not build any structure. The serial integration mechanism is responsible for building new structures, but is constrained to maintaining a small number of constructed representations, in parallel, in working memory which is composed of ACT-R buffers supplemented with specialized language analysis buffers (Ball, 2011b).

Cognitive processing in ACT-R revolves around the selection and execution of a sequence of productions. The production with the highest utility that matches the current context provided by the ACT-R/language analysis buffers, is selected for execution. Production execution can result in a perceptual-motor action (e.g. visual attention shift, mouse movement), a modification to the contents of a buffer, or a DM retrieval. These actions change the context for selection and execution of the next production.

When the executing production invokes a DM retrieval, the parallel spread of activation from chunks in buffers to associated chunks in DM (*soft constraints* or biases) combines with the base level activation—based on prior history of use of the chunk—to determine total chunk activation. The single most highly activated chunk which matches a retrieval template (*hard constraint*) specified by

¹ Thanks to Dan Bothell for pointing out several misconceptions about ACT-R in an earlier version of this paper

the executing production is retrieved. Chunks can either be associated by sharing a slot value or by explicit specification of an association using the `add-sji` function. For activation to spread, the activating chunk must be in a buffer (matching slot value) or in a slot in a chunk in a buffer (explicit specification via `add-sji`).

The language analysis model makes extensive use of ACT-R's serial and parallel processing mechanisms. The model processes the linguistic input incrementally, one word or multi-word unit at a time, and uses all available information interactively to make the best choice at each choice point. The model also relies on a non-monotonic mechanism of *context accommodation* which is capable of making modest adjustments to the evolving representation when the current input, in combination with the current context, indicates the need for such accommodation. Context accommodation is part of normal processing—in the right context, a production capable of accommodating the input executes. For example, in incrementally processing “the airspeed restriction”, when “airspeed” is processed, it is integrated as the head of the noun phrase projected during the processing of “the”, but when “restriction” is subsequently processed, the model accommodates “restriction” by shifting “airspeed” into a modifier function and making “restriction” the head. Context accommodation is not capable of handling the kinds of disruptive garden path sentences that are a mainstay of psycholinguistic research (e.g. Bever's (1970) famous “the horse raced past the barn fell”). Such inputs require reanalysis mechanisms which have not yet been implemented. The focus of model development is on handling common English—inputs which humans process with ease, but which, nonetheless, present significant modeling challenges due to ambiguity. The combination of parallel/probabilistic/context dependent processing, and serial processing with context accommodation allows the model to pursue the single best analysis, but to adjust the analysis without backtracking or reanalysis, when needed. The overall result is a *pseudo-deterministic* HLP which presents the appearance and efficiency of deterministic processing, despite the rampant ambiguity which makes truly deterministic processing impossible (Ball, 2011a).

Activation

In ACT-R, all DM chunks have an activation level which depends on the current context (source activation) and prior history of use (base level activation) of the chunk. A key assumption is that the current context is captured in the contents of the ACT-R buffers which are sources of activation. The most basic form of the activation equation (ignoring partial matching which we do not use, and noise) is shown below where A_i = total activation of chunk i ; B_i = base level activation of chunk i , and S_i = spreading activation contribution to activation of chunk i :

$$A_i = B_i + S_i$$

The base level activation is a logarithmic function of the number of uses of a chunk over time combined with a

negative exponential decay mechanism (assuming the default, optimized base level equation). Spreading activation is a weighted sum of activations from all the sources of activation in buffers which match the slot values of the chunk being activated or for which an explicit association has been specified (via `add-sji`). The amount of spreading activation to a chunk from each source decreases with the number of competing chunks which match the source. This proportional spreading activation is known as the *fan effect*. The fan effect does not apply to chunks for which an explicit association has been specified.

The language analysis model makes extensive use of ACT-R's retrieval (activation and selection) mechanism. In the word recognition subcomponent, a perceptual span which encodes the visual contents of the current attention fixation spreads activation to DM and the word or multi-word unit which is most highly activated is retrieved (selected) and compared to the perceptual input. If the comparison is close enough, the retrieved word or multi-word unit is considered a match. Overall, the process involves four steps: 1) perceptual encoding of the input (*encoding*); 2) activation of declarative memory (*activation*); 3) retrieval of the most highly activated DM chunk which matches the hard constraints of the retrieval template (*selection*); and 4) comparison of the retrieved memory element against the perceptual input (*verification*). Completion of all but the third step presents challenges for ACT-R based modeling.

ACT-R's built in perceptual encoding mechanism assumes words are divided into units by spaces and automatically separates punctuation into separate perceptual units. While this typically succeeds in identifying words and punctuation, it often does not. There are words like “etc.” and “didn't” which incorporate punctuation and there are words like “a priori” and “none the less” which have spaces. In addition, the model includes multi-word units like “have been”, “get out” and “New York” which are encoded in the mental lexicon as lexical items. Higher level knowledge from the mental lexicon is needed to decide what constitutes a word or multi-word unit. To support the integration of higher level knowledge with perceptual processing, we modified ACT-R's perceptual encoding mechanism to incorporate a perceptual span that does not automatically segment the input at spaces and punctuation (Freiman & Ball, 2010). Not only does the perceptual span mechanism support the integration of higher level knowledge, it speeds up processing significantly since words like “don't” are recognized as a single unit instead of three separate units “don”, “'” and “t”. Likewise, multi-word units like “get out” are also recognized as a unit. Besides speeding up processing, the recognition of multi-word units reduces ambiguity significantly. The word “take” is extremely ambiguous, whereas multi-word units like “take out”, “take off” and “take in” are far less ambiguous. The ability to recognize multi-word expressions is an important tool for handling the ambiguity of natural language and for speeding up the model. We view the addition of multi-word expressions as the best way of achieving adult reading rates.

With a mental lexicon near 58,000 lexical items, the computation of activation presents a serious computational challenge. It is not possible to compute the activations of 58,000 lexical items prior to each retrieval, in real-time, on existing hardware. (This is also the reason we are unable to use the partial matching subsystem, since all DM chunks are candidates for retrieval when partial matching is enabled.) As a workaround, we developed a capability to minimize the activation computations in the event of an exact match to the form of the input. If there is a lexical item in DM which is an exact match to the perceptual span, a hard constraint is added to the retrieval template to restrict the number of matching DM elements. When the full perceptual span doesn't match, the match is backed off to the last space in the perceptual span and re-attempted. Prior spaces can also be backed off to. If there is no match (e.g. if the input is "spped")—as a computational compromise—the model attempts a retrieval requiring a hard constraint match on the first letter in the perceptual span. We call this mechanism a *disjunctive retrieval* capability. Except for this last compromise, the disjunctive retrieval capability retrieves the same lexical item as a soft constraint retrieval. Even with this last compromise, computation of activations is slower than real-time in the worst case where only a first letter match is required, since there may be thousands of matching lexical items whose activation must be computed. We are looking for ways to improve processing with minimal compromise compared to the preferred soft constraint retrieval mechanism.

The verification step is also problematic from an ACT-R modeling perspective. ACT-R does not provide the kind of low level perceptual matching capability that is needed to implement this step. Instead, we have incorporated the Levenshtein Distance algorithm to perform this comparison. We view verification as a key element of the word recognition mechanism in accord with the Activation-Verification model of Paap et al. (1987) and in contrast to the Interactive-Activation model of McClelland & Rumelhart (1985) which has no verification stage. Verification is crucial for identifying novel inputs. A novel input is one that is not a close match to any chunk in memory, although exactly what constitutes a "close match" is an open research question.

Multi-Level Activation Spread

In ACT-R, activation spreads from the slots in chunks in buffers to chunks in DM with matching slot values or explicitly set associations (using **add-sji**). For example, we have a context buffer that encodes information about the context that has a slot named "gram-pos-bias" (grammatical part of speech bias). Following the processing of a word like "the" (a determiner), this slot will be set to the chunk **noun**. During the retrieval of a lexical item, the **noun** chunk will spread activation to all lexical items with a matching **noun** chunk (i.e. all nouns). If the word "point" follows "the", this bias will spread activation to the **noun** chunk for "point" as opposed to the **verb** chunk (i.e. "to point"). In this way the

grammatical context biases the selection of the part of speech (POS) of a word during retrieval.

There is no mechanism in ACT-R to spread activation from slots in chunks in DM to other chunks in DM with matching slot values or to explicitly associated chunks. Once activation spreads from slots in buffers to DM chunks during a retrieval, activation spread stops and the final activation is computed to determine which DM chunk to retrieve. We refer to this as *single level activation spread*.

Our model assumes that there are DM chunks which encode both the form of a word (e.g. "speed", "speeds") and POS (e.g. "noun", "verb"). Originally, word form and POS information were encoded in distinct **word-form** and **pos** chunks. The model first retrieved a **word-form** chunk given the letters and trigrams in the input, then retrieved a **pos** chunk for the word form. In order to improve the analysis rate of the model (Freiman & Ball, 2010), word form and part of speech information was combined into a single **word-pos** chunk (i.e. word form + part of speech). While we were successful in eliminating a retrieval, the resulting **word-pos** chunks contain a mixture of word form information (e.g. the letters and trigrams in the word) and POS information (e.g. noun, verb, as well as grammatical features like number, animacy and gender for nouns, and tense and aspect for verbs). Note that this mixture of word form and POS information makes it possible to capture the interaction of word form and POS with single level activation spread. For example, retrieval of the POS for "speed" (i.e. noun or verb) given the input "spped" depends on the biasing context (e.g. noun bias following "the", verb bias following "to") as well as the letters and trigrams. However, the **word-pos** chunks do not (yet) contain any representation of phonetic, phonemic, syllabic or morphemic information. With just letter, trigram and POS information, **word-pos** chunks contain many slots. Adding phonetic, phonemic, syllabic and morphemic information will increase the number of slots substantially. Ideally, we would like to represent letter, trigram, POS, phonetic, syllabic etc. information independently of each other in separate chunks—allowing them to interact in retrieving a word (or letter, or POS, or phoneme), but given the single-level activation spreading mechanism in ACT-R, there is no way to capture the interaction without including all the linguistic knowledge in a single chunk or using **add-sji** to establish links between chunks and retrieving and retaining chunks at all levels in buffers to spread activation.

A negative consequence of the integration of word form and POS information is the need to redundantly encode information. If a given word form is associated with multiple POSs, then multiple **word-pos** chunks are needed. For example, the word "speed" can be both a noun and a verb. To represent this, two **word-pos** chunks are needed, one for the noun and one for the verb. In these two **word-pos** chunks, the letters and trigrams in the word form are redundantly encoded. To minimize redundancy, it is desirable to factor out word form and POS knowledge. But to model adult human reading rates, it is important to minimize the number of retrievals. Both can be

accomplished with multi-level activation spread. For example, if the goal is to retrieve a POS, the letters and trigrams in the input can spread activation to a **word-form** chunk which can spread activation to a **pos** chunk. Then the **pos** chunk can be retrieved without first retrieving the **word-form** chunk.

We are in the process of mapping the linguistic representations that are generated by our language analysis model into a situation model based semantic representation. We are trying to do this in a representationally reasonable way within ACT-R. The problem we face is the many-to-many mapping between words and concepts. Individual words may map to multiple concepts (river “bank” vs. financial “bank”), and individual concepts may map to multiple words (“dog” vs. “canine”). Given this many-to-many mapping, we would like to use **mapping** chunks to map from words to concepts. The **mapping** chunks would encode a single mapping relationship (e.g. a separate mapping chunk to map from the word “bank” to the financial institution concept; from the word “bank” to the river bank concept; from the concept dog to the word “dog”; from the concept dog to the word “canine”). When processing a word, a key goal is to retrieve the contextually relevant concept. We would like to accomplish this with a minimum number of retrievals since our model is already slower than adult humans even without the mapping into concepts. Since there is no direct link between a word and a concept if **mapping** chunks are used (i.e. there is no slot in the **concept** chunk that contains the word), the word will not spread activation to the concept. Instead, given the use of **mapping** chunks, two retrievals are needed: 1) given a **word-pos** chunk, retrieve a **mapping** chunk, and 2) given a **mapping** chunk, retrieve a **concept** chunk. The use of **mapping** chunks can be eliminated if we use the **add-sji** function to establish direct links between **word-pos** chunks and **concept** chunks. We are currently pursuing this option to avoid the need to retrieve an intermediate **mapping** chunk. Even with explicit links from **word-pos** chunks to **concept** chunks, a **word-pos** chunk must first be retrieved to spread activation to associated **concept** chunks. With multi-level activation spread it would be possible to directly retrieve a **concept** chunk, eliminating the need to retrieve a **word-pos** chunk.

Alternatively, if we were to combine **concept** chunks with **word-pos** chunks, then a single retrieval could be used to retrieve a **word-pos-concept** chunk. However, there may be multiple concepts associated with a **word-pos** chunk (e.g. “river *bank*” vs. “financial *bank*”). If we create separate **word-pos-concept** chunks for each alternative, the amount of redundancy is increased again. Further, it is questionable whether letter and trigram information should be directly associated with (non-linguistic) concepts.

The main advantage of creating **word-pos-concept** chunks is the reduction in the number of retrievals needed to go from the input to a concept. To see how problematic retrievals are for models of reading, consider the E-Z Reader model (Reichle, Warren & McConnell, 2009), a

model of eye movements in reading which models lexical processing (not reading). E-Z Reader allows just 25 msec per word beyond lexical access for post-lexical processing to influence lexical processing. According to the authors, 25 msec is “the minimal amount of post-lexical processing that (on average) is necessary to satisfy the language-processing system that comprehension is proceeding without difficulty and that it is not necessary to interrupt lexical processing and/or halt the progression of the eyes” (ibid., p. 6). Since it requires 50 msec to execute a production in ACT-R which attempts a retrieval, plus the retrieval time, there would be insufficient time for a single retrieval in an ACT-R implementation of E-Z Reader to influence lexical processing and eye movements! However, the E-Z Reader model makes the simplifying assumption that words are space delimited and since our model is capable of recognizing multi-word units, the 25 msec limit can be relaxed somewhat. But there is still insufficient time for more than 1 or 2 retrievals (on average) beyond the retrieval needed to support word recognition itself.

Carry Over Activation and Resonance

Activations are computed in ACT-R as part of a retrieval attempt. The activation computation involves combining the base level activation and activation spread from all buffers which are sources of activation. The logarithmic nature of the default, optimized base level activation computation means that the base level of overused DM chunks does not vary much from use to use (i.e. the base level activation has reached asymptote). Words constitute very highly used DM chunks. Using estimates of the number of occurrences of a word over a lifetime results in a base level activation that varies little from use to use and decays very slowly. Since the spread of activation is computed independently on each retrieval, for a word that has been used recently, there is no contextual indication of this prior use (i.e. the base level hasn’t significantly changed and any prior spread of activation is not retained). Yet there is clear evidence of priming effects from prior uses of words. According to Dan Bothell (p.c.), this is a limitation of ACT-R’s default optimized base level equation. We are exploring use of a hybrid version of the base level equation which does not suffer this limitation (Bothell, 2011, p. 213). Use of the non-optimized base-level equation is not possible given the size of our mental lexicon and the large number of word uses.

Another alternative is to retain the word in a buffer so that activation can continue to spread from the word to corresponding chunks in DM. We have tried this approach in the case of idiom processing. To see the basic challenge, consider the processing of idioms like “kicked the bucket” and verb-particle combinations like “pick...up” as in “pick the ball up”. We assume that idioms and verb-particles correspond to distinct chunks (i.e. multi-word units) in DM. These multi-word expressions exceed the size of the perceptual span and cannot be recognized in a single attention fixation. Instead, the model must somehow recognize the idiom “kicked the bucket” when the word “bucket” is processed and the verb-particle combination

“pick...up” when the word “up” is processed. If there is no evidence that “kicked” has occurred at the processing of “bucket”, then there is no way for the model to retrieve “kicked the bucket” instead of “bucket”. Similarly for “pick” when “up” is processed. Since the DM element “bucket” is an exact match to “bucket” and “bucket” has a higher base frequency than “kicked the bucket” (i.e. single words have a higher base frequency than multi-word units containing them), there must be some mechanism for preferring “kicked the bucket” in this context. “Kicked” and “the” could be retained in the context to spread activation to “kicked the bucket” to handle this example, but, in general, this would mean retaining an arbitrary number of words in the context to spread activation. In the case of “pick...up”, “pick” would need to be retained in a buffer for an indefinite period of time (e.g. “*pick* the big red ball *up*”, “*pick* the ball that is on the table *up*”).

Even with the hybrid base-level equation, relying on an increase in base level for “kicked the bucket” will not work in this example. Since the processing of “kicked” is likely to retrieve “kicked” and not “kicked the bucket”, the base level activation of “kicked the bucket” will be unaffected at the processing of “kicked” (i.e. the “kicked the bucket” chunk must be retrieved and merged back into DM to constitute a use). Further, any temporary spreading activation from “kicked” to “kicked the bucket” will have been lost at the processing of “bucket”.

A possible solution is to introduce a carry-over activation capability. When “kicked” is processed it will spread activation to “kicked the bucket” as well as “kicked”. Despite the fact that “kicked the bucket” is not retrieved, some of this activation will carry-over so that when “bucket” is processed, “kicked the bucket” will receive activation from “bucket” as well as carry-over activation from “kicked” and “the”. The combination of carry-over activation from “kicked” and “the”, plus the activation from “bucket” should allow “kicked the bucket” to be retrieved in this context. In general, this seems like a better solution than trying to retain “kicked” and “the” in the context when “bucket” is processed. In the case of “pick...up”, carry over activation should handle cases where the gap between “pick” and “up” is small (e.g. “*pick* the ball *up*”), but cause problems when the gap is large enough that any carry over activation will have decayed. This result might explain the preference for placing the particle before the object when the description of the object is long (e.g. “*pick up* the big red ball on the table” is preferred over “*pick* the big red ball on the table *up*”).

There are additional reasons for suggesting the introduction of carry-over activation. Carry-over activation corresponds to a short-term increase in the activation of a DM chunk that extends beyond the execution of a single chunk retrieval. This carry-over activation (i.e. neuron spiking) differs from increases in base level activation which we view as more permanent changes in long-term potentiation. The introduction of carry-over activation combined with multi-level activation spread, could support an ART-like adaptive resonance capability (Grossberg,

1987)—although it is unclear how this could be done in a computationally tractable way. Note that ART uses resonance to distinguish novel from previously experienced inputs—previous inputs lead to resonance with memory whereas novel inputs do not. With carry-over activation, the verification stage of the word recognition subcomponent could be implemented within the architecture via resonance instead of using the Levenshtein Distance metric outside the architecture.

Inhibition

Inhibition is a winner-take-all mechanism that is commonly used in connectionist architectures to allow a network of nodes to settle into a solution (cf. McClelland & Rumelhart, 1981; Kintsch, 1998). Over time, the most active node or co-activating nodes inhibit competing nodes. There is no equivalent in ACT-R—although it is possible to get inhibitory effects by explicitly setting the strength of association between two or more chunks to a negative value. But even here, there is no notion of settling into a solution in ACT-R.

The need for inhibition as a mechanism for settling into a solution is obviated in ACT-R by the retrieval mechanism which results in selection of the single most highly activated chunk matching the retrieval template. This is ACT-R’s equivalent of a “winner-take-all” network. The retrieval mechanism picks out the most highly activated chunk which matches the hard constraints of the retrieval template.

ACT-R’s spreading activation mechanism doesn’t bias a model to any particular task. The same cannot be said of inhibition. For any inhibitory network, it is possible to define conflicting tasks that the inhibitory network cannot perform. For example, if both singular and plural forms of nouns (e.g. “child” and “children”) occur in a network, should they inhibit each other? It depends on the task. If the task is a lexical decision task, then we want “child” to inhibit “children” and vice versa, so that they don’t interfere (i.e. if “child” is the winner when the input is “children”, presumably the lexical decision response would be negative since “child” doesn’t match the input). On the other hand, if the task is to generate the singular form of the word in response to the plural word, or the plural in response to the singular, then we need facilitation rather than inhibition. As another example, consider the verbs “go” and “went”. For a lexical decision task, these words should inhibit each other. But for a task of generating the past tense of “go”, they should facilitate each other.

Not only are inhibitory links task specific, but in a large declarative memory, the number of such links will be explosive. If a word must inhibit all its competitors, with a 58,000 word lexicon, the number of inhibitory links is computationally explosive. Inhibitory links don’t scale.

Besides the task specificity of inhibitory networks, most inhibitory networks assume well-defined levels with inhibitory links typically constrained to occurring within a level and excitatory links occurring across levels. The recognition of multi-word expressions like “get up”, inflected words like “books” and morphologically complex

words like “progressivity” present a challenge for such networks? Are multi-word expressions (e.g. “get up”), inflected words (e.g. “books”), morphologically complex words (e.g. “progressivity”), morphologically simple words (e.g. “cat”) and morphemes (e.g. plural “s”) represented on the same level, in which case they compete, or on different levels, in which case they co-activate each other? It depends on the task. If “get”, “up” and “get up” are all represented on the same *lexical item* level where they inhibit each other, how do we recognize “get up” as a lexical unit (i.e. how does the model settle in to “get up”)? Even if multi-word expressions are represented on a different level from words, the words in the multi-word expression will inhibit each other, making it difficult to distinguish the multi-word expression from the word which wins the word level competition, unless the task is specifically to recognize multi-word expressions. Similar questions arise for inflected, morphologically complex, and morphologically simple words, and morphemes. In short, multi-word expressions, inflected words, morphologically complex words and morphemes call into question the typical assumption that there is a well defined word level. In a model restricted to four letter words without inflectional variants where the task is word recognition (e.g. McClelland & Rumelhart, 1981), well-defined levels can be established. When we consider real language, there is no well-defined word level with inhibitory links that is task independent.

In sum, inhibition is not a viable alternative to ACT-R’s task general spreading activation mechanism combined with a winner-take-all retrieval mechanism that depends on the current task.

Conclusions

The use of ACT-R for language analysis provides several benefits. ACT-R solves the problems of how to integrate symbolic and probabilistic processing combined with serial and parallel processing in an effective and elegant manner. For the most part, the capabilities provided by ACT-R have proved useful for the development of our language analysis model, and much of the success of our model is attributable to the capabilities and constraints of ACT-R.

However, there is room for improvement of ACT-R. Interestingly, the suggestions presented in this paper are consistent with Anderson’s seminal paper on spreading activation (Anderson, 1983a) and the ACT* architecture (Anderson, 1983b). Multi-level activation spread is capable of spreading activation to indirectly related DM chunks, obviating the need to specify indirect links or to add slots to directly model the associations, and keeping the capability to model interactions and minimize retrievals. Carryover activation allows the effects of multi-level spreading activation to be retained until needed. Of course, the computational costs of multi-level activation spread and carry-over activation are potentially explosive and it will be a challenge to figure out how to extend ACT-R’s capabilities in the ways suggested in this paper in a computationally tractable manner.

References

- Altmann, G., & Steedman, M. (1988). Interaction with context during human sentence processing. *Cognition*, *30*, 191-238.
- Anderson, J. (1983a). A Spreading Activation Theory of Memory. *Journal of Verbal Learning and Verbal Behavior*, *22*, 261-295.
- Anderson, J. (1983b). *The Architecture of Cognition*. Cambridge, MA: Harvard University Press.
- Anderson, J. (2007). *How Can the Human Mind Occur in the Physical Universe?* NY: Oxford University Press.
- Ball, J. (2003). Beginnings of a Language Comprehension Module in ACT-R 5.0. *Proceedings of the Fifth International Conference on Cognitive Modeling*. F. Detje, D. Doermer & H. Schaub (eds). Univ-Verlag Bamberg. ISBN 3-933463-15-7.
- Ball, J. (2011a). A Pseudo-Deterministic Model of Human Language Processing. *Proceedings of the 2011 Cognitive Science Society Conference*.
- Ball, J. (2011b). Explorations in ACT-R Based Cognitive Modeling – Chunks, Inheritance, Production Matching and Memory in Language Analysis. *Proceedings of the AAAI Fall Symposium: Advances in Cognitive Systems*, 10-17. Arlington, VA: AAAI.
- Ball, J., Freiman, M., Rodgers, S. & Myers, C. (2010). Toward a Functional Model of Human Language Processing. *Proceedings of the 32nd Annual Meeting of the Cognitive Science Society*.
- Ball, J., Heiberger, A. & Silber, R. (2007). Toward a Large-Scale Model of Language Comprehension in ACT-R 6. In R. Lewis, T. Polk & J. Laird (Eds.) *Proceedings of the 8th International Conference on Cognitive Modeling*. 173-179. NY: Psych Press.
- Bever, T. (1970). The cognitive basis for linguistic structures. In J. Hayes (ed.), *Cognition and the development of language*, 279-362. NY: Wiley.
- Bothell, D. (2011). ACT-R 6.0 Reference Manual. Downloaded from <http://act-r.psy.cmu.edu/actr6/reference-manual.pdf>
- Freiman, M. & Ball, J. (2010). Improving the Reading Rate of Double-R-Language. In D. D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the 10th International Conference on Cognitive Modeling* (pp. 1-6). Phil, PA: Drexel University.
- Gibson, E., & Pearlmuter, N. (1998). Constraints on sentence comprehension. *Trends in Cognitive Sciences*, *2*(7), 262-268.
- Grossberg, S. (1987). Competitive Learning: from interactive activation to adaptive resonance. *Cognitive Science*, *22*, 23-63.
- Just, M. & Carpenter, P. (1987). *The Psychology of Reading and Language Comprehension*. Boston: Allyn and Bacon, Inc.
- Kintsch, W. (1998). *Comprehension: A paradigm for cognition*. NY: Cambridge University Press.
- McClelland, J., & Rumelhart, D. (1981). An interactive activation model of context effects in letter perception: I. An account of basic findings. *Psychological Review*, *88*(5), 375-407.
- Paap, K., Newsome, S., McDonald, J., & Schvaneveldt, R. (1982). An Activation-Verification Model of Letter and Word Recognition: The Word-Superiority Effect. *Psychological Review*, *89*, 573-594.
- Reichle, E., Warren, T. & McConnell, K. (2009). Using E-Z Reader to model the effects of higher level language processing on eye movements during reading. *Psychonomic Bulletin & Review*, *16*(1), 1-21.
- Tanenhaus, M., Spivey-Knowlton, M., Eberhard, K., & Sedivy, J. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, *268*, 1632-1634.

Prospective Memory and Working Memory: A Hierarchical Modeling Approach

Nina R. Arnold (nina.arnold@hhu.de)

Institute of Experimental Psychology, Heinrich-Heine-Universität Düsseldorf,
Universitätsstr. 1, D-40225 Düsseldorf, Germany

Ute J. Bayen (ubayen@hhu.de)

Institute of Experimental Psychology, Heinrich-Heine-Universität Düsseldorf,
Universitätsstr. 1, D-40225 Düsseldorf, Germany

Rebekah E. Smith (rebekah.smith@utsa.edu)

Department of Psychology, University of Texas at San Antonio,
One UTSA Circle, San Antonio, TX 78249, U.S.A.

Keywords: beta MPT; multinomial modeling; prospective memory; working memory.

model has four free parameters P , M , C_1 , and C_2 , and is identifiable.

Prospective Memory and Working Memory

Prospective memory (PM), remembering to perform an action in the future, is an important ability in everyday life. It involves a prospective component, remembering *that* you have to do something, and a retrospective component, remembering *what* action to perform and *when* to perform it. Researchers have pointed to a role of cognitive factors, such as individual differences in working memory (WM), in determining PM performance. Previous studies have found a positive relationship between WM span and PM (e.g., R.E. Smith, Persyn, & Butler, 2011). The goal of the present study is to apply a hierarchical modeling approach to investigate how WM is related to the prospective and retrospective components of PM.

The Multinomial Model of Event-Based PM

PM tasks often involve interrupting an ongoing activity. Therefore, laboratory PM tasks are often embedded in an ongoing task. Smith and Bayen (2004) introduced a multinomial model of event-based PM that can be applied to PM tasks that are embedded in an ongoing task with two response options, such as a lexical decision task. While participants are engaged in the ongoing task, they are supposed to execute a specific action when a PM target occurs. The model (Figure 1) includes parameter P which measures the prospective component and parameter M which measures retrospective recognition memory processes for discriminating PM targets and nontargets. Parameter g defines the probability of guessing that a PM target is present. C_1 and C_2 measure processes related to the ongoing task. For example, in a lexical decision task, C_1 is the probability of correctly detecting that a letter string is a word and C_2 is the probability of correctly detecting that a letter string is a non-word. If a participant does not detect that a string is a word or a non-word, he or she guesses with probability c that it is a word. It is assumed that participants adjust their responses to the perceived ratio of items in a task, that is $c =$ the ratio of word trials to total trials and $g =$ the ratio of PM target trials to total trials. The resulting

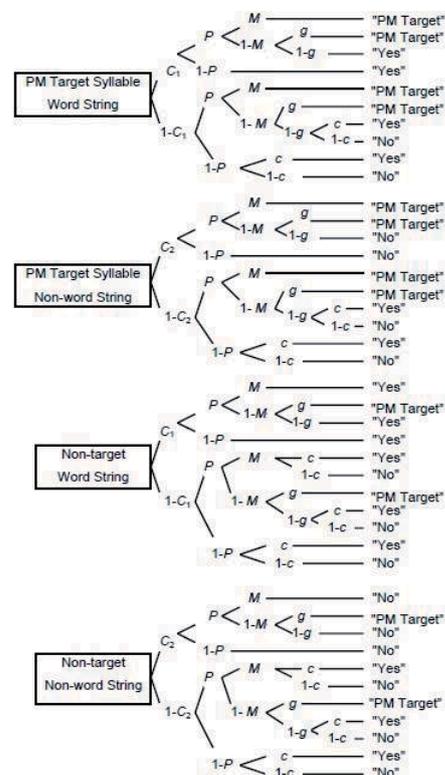


Figure 1: The Multinomial Model of Event-Based PM. Adapted from “A multinomial model of event-based prospective memory” by R.E. Smith and U.J. Bayen, 2004, *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 30, p. 758.

Beta Multinomial Processing Tree Models

Since multinomial models are often applied to data that have been aggregated over participants and items, they assume that all observations are independent and identically distributed and ignore differences between participants and items (Riefer & Batchelder, 1988). However, J.B. Smith and Batchelder (2008) showed that this assumption is often

violated even for a carefully constructed item pool and relatively homogeneous groups of participants. This can result in biased parameter estimates.

The beta MPT (J. B. Smith & Batchelder, 2010) assumes that each participant's parameters are drawn independently from a multivariate distribution consisting of independent marginal beta-distributions. The advantage of the beta distribution is that it lies in the interval [0,1] and thus in the natural parameter space of the model parameters because they represent probabilities. Beta-MPT models are computed using Markov Chain Monte Carlo (MCMC) methods. This analysis is possible without having to solve the integrals which can be computationally very expensive. It can be easily applied using software like WinBUGS (Spiegelhalter, Best, & Lunn, 2003).

Study by R. E. Smith, Persyn, & Butler (2011)

413 participants took part in this experiment. The ongoing task was a lexical decision task. The PM task was to press the F1 key when syllables “low” and “per” appeared. Participants completed a symmetry span task as a measure of WM span. Participants whose WM score fell within the lowest 25% were placed into the lower WM group and those whose WM score was in the top 25% were placed in the higher WM group. In R. E. Smith et al.'s (2011) analysis, participants in the higher WM group had a higher probability of remembering that something needed to be done (the prospective component measured by model parameter P), but the two WM groups did not differ on the retrospective component as measured by parameter M .

Reanalysis

The extreme-group analysis is limited in that half of the data were omitted. By using the beta-MPT approach we are able to incorporate data from all participants to address the question whether and how individual differences in working memory span contribute to successful PM.

For each Beta-MPT we used a uniform distribution between 1 and 5000 as prior for α and β . This is a very vague prior because of its wide range, but because α and β are greater than 1, it ensures that the beta-distribution is bell-shaped. We used 100,000 iterations and discarded the first half of the iterations as burn-in period.

With the resulting individual parameter estimates, we computed correlations and a regression analysis. The correlation of WM span and the prospective component P was significant, $r = .15$, $p < .01$, but WM span and the retrospective-memory component M were not associated with one another, $r = .01$, despite sufficient power to detect even a small effect. In the regression analyses, when only WM span was entered as predictor variable for PM performance it was a significant predictor. When P was entered as an additional predictor variable, WM span was not a significant predictor. The inclusion of M as predictor had no influence on the predictive value of WM span.

Table 1: Results of regression analyses for criterion variable PM performance.

Parameter	b	SE	β	t	p	R^2
Constant*	0.41	0.07		6.24	<.01	.01
WM*	0.01	<0.01	.11	2.32	.02	
Constant*	-0.24	0.03		-8.10	<.01	.85
WM	<0.01	<0.01	-.02	-1.15	.25	
P^*	1.22	0.03	.92	47.24	<.01	
Constant*	-0.63	0.01		-53.93	<.01	.98
WM	<0.01	<0.01	.01	0.79	.43	
P^*	0.93	0.01	.70	94.78	<.01	
M^*	0.72	0.01	.43	58.43	<.01	
Constant*	-0.59	0.06		-10.56	<.01	.62
WM*	0.01	<0.01	.11	3.45	<.01	
M^*	1.31	0.05	.78	25.80	<.01	

Thus, the results match previous findings. WM span is related to the prospective component of PM, but not to the retrospective-memory component. Beta-MPT models enabled us to incorporate data from all participants, to look at the data on an individual basis, and to avoid biased parameter estimates. They provide information about the relationship between individual differences in WM span and cognitive processes underlying PM. Regression analyses showed that the relationship between WM span and PM performance was fully accounted for by the prospective component P .

Acknowledgments

The research was supported by Grant AG034965 from the National Institute on Aging.

References

- Riefer, D. M., & Batchelder, W. H. (1988). Multinomial modeling and the measurement of cognitive processes. *Psychological Review*, *95*, 318–339.
- Smith, J. B., & Batchelder, W. H. (2008). Assessing individual differences in categorical data. *Psychonomic Bulletin & Review*, *15*, 713–731.
- Smith, J. B., & Batchelder, W. H. (2010). Beta-MPT: Multinomial processing tree models for addressing individual differences. *Journal of Mathematical Psychology*, *54*, 167–183.
- Smith, R. E., & Bayen, U. J. (2004). A multinomial model of event-based prospective memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *30*, 756 - 777.
- Smith, R. E., Persyn, D., & Butler, P. (2011). Prospective Memory, Personality, and Working Memory: A Formal Modeling Approach. *Zeitschrift für Psychologie / Journal of Psychology*, *219*, 108–116.
- Spiegelhalter, D. J., Thomas, A., Best, N., & Lunn, D. (2003). WinBUGS user manual version 1.4. Cambridge, UK: MRC Biostatistics Unit.

Non Inversion of Similarity and Dissimilarity in Judging Faces

Ludovica Lorusso (lorusso@uniss.it)

Department of Economics
Via Muroli, 25, Sassari, 07100 Italy

Enrico Grosso (grosso@uniss.it)

Department of Political Science, Communication and Computer Engineering
Viale Mancini, 5, Sassari, 07100 Italy

Keywords: Visual perception; face stimuli, similarity, kinship.

We propose a model for three judgments involving a measure of similarity and dissimilarity between faces: the similarity judgment (Sj), the dissimilarity judgment (Dj), and the kinship judgment (Kj), which has been demonstrated to be strongly related to the Sj (Lorusso et al., 2011; Maloney & Dal Martello, 2006).

A previous model for Sj and Kj between faces (see i.e., Maloney & Dal Martello, 2006) suggests a common visual pathway for both the judgments where a similarity measure of *similarity cues* also named “kin signals” is required as a necessary step to judgment of both similarity and kinship. DeBruine et al. (2009) found that the observed correlation between Sj and Kj *depends on the face stimulus presented*: whenever face-pairs differ in age or sex, a similarity measure is not found as a criterion of kinship evaluation. The model we present here is based on the results obtained in a recent work (Lorusso et al., 2011). These results showed significant differences in response time for Sj, Dj, and Kj on different face-pairs categories previously defined on the base of an experiment of similarity ratings. Participants took on average about half a second less to respond to stimulus pairs rated as dissimilar than to either those rated as similar or those exhibiting kinship. Moreover, on average participants took about a third of a second longer to judge kinship than to judge similarity. A slight difference was also found between Kj and Dj. Finally, a priming study showed a strong priming of Sj on Kj and vice versa. A strong priming was also observed of Dj on both Sj and Kj. However, the priming effect was suppressed whenever Sj and Kj were given before Dj: in this case, for example, a positive Dj followed a negative Sj and Kj with a chance level frequency. These results suggest a new scenario in which judgments of similarity, dissimilarity and kinship of faces are modulated by both the task and the stimulus and where different visual and cognitive pathways are involved during each of them. Moreover, Sj and Dj cannot be considered in a logical opposition between each other. Our model - sketched in Figure 1 - suggests that any specific task leads the observer to process specific pools of facial features and use that information in order to complete the judgment. The fact that Sj and Dj show a non inversion is explained in the model by hypothesizing that they rely on a processing of different pools of visual information, respectively the similarity and dissimilarity pools in Figure 1.

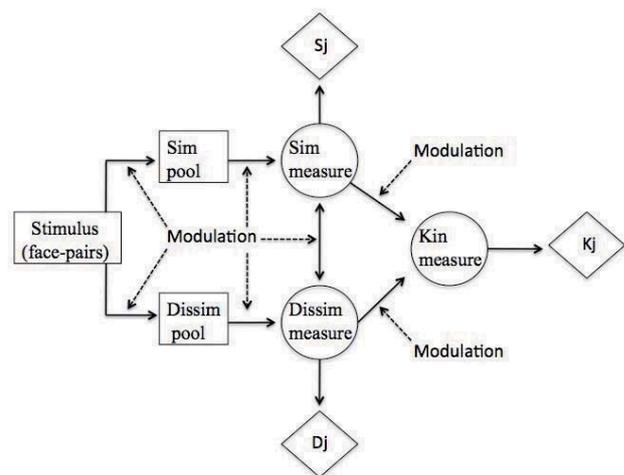


Figure 1: The new model for similarity, dissimilarity and kinship judgments of faces.

The model assumes that Sj and Dj occur sooner than Kj since they only require a very simple processing of the two pools: they may be therefore reduced to elementary similarity/dissimilarity measures. Kj is the slowest since it involves a more complex processing of the two pools, and therefore it cannot be reduced to a simple similarity measure of similarity cues. In the model the similarity measure represents only one of the possible strategies used in a complex judgment as a Kj - likely the strategy adopted in the case of faces perceived as highly alignable like faces matching for sex and age. Finally, the strong correlation between Kj and Sj observed in the priming study may be explained to be part of a cultural modulation that we express through our anecdotal association of the concept of kinship with similarity.

References

- DeBruine, L. M., Smith, F., Jones, B., Roberts, S., Petrie, M., & Spector, T. (2009). Kin recognition signals in adult faces. *Vision Research*, 49, 38-43.
- Lorusso, L., Brelstaff, G., Brodo, L., Lagorio, A., & Grosso, E. (2011). Visual judgments of kinship: An alternative perspective. *Perception*, 40, 1282-1289.
- Maloney, L. T., & Dal Martello, M. F. (2006). Visual judgments of kinship: An alternative perspective. *Journal of Vision*, 6, 1047-1056.

Towards optimal payoff manipulations

Christian P. Janssen¹, Andrew Howes², and Duncan P. Brumby¹

¹UCL Interaction Centre, University College London
London WC1E 6BT UK

²School of Computer Science, University of Birmingham
Birmingham, B15 2TT UK

c.janssen@ucl.ac.uk, howesa@bham.ac.uk, d.brumby@cs.ucl.ac.uk

Keywords: Payoff function, mathematical modeling, performance trade-offs, cognitively bounded rational analysis

Introduction

Multitasking typically requires people to make performance trade-offs: paying more attention to one task can improve performance there, but might lead to performance decrements on other unattended tasks. In our work we try to gain a better understanding of how people make such trade-offs. One difficulty in this effort is that performance is typically expressed in different units across tasks (e.g., “accuracy” of keeping a car inside a lane and “speed” of performing a secondary task such as dialing). How do people trade-off these different units?

Explicit payoff functions have been proposed as a way to achieve the desired trade-off (e.g., Howes, Lewis, & Vera, 2009; Janssen, Brumby, Dowell, Chater, & Howes, 2011; Payne, Duggan, & Neth, 2007; Schumacher, et al., 1999). They can be used to translate performance on multiple tasks into a single score. The participant and the modeler can then use this feedback to objectively compare performance for different strategies (Howes, et al., 2009; Janssen, et al., 2011). If successful, payoff functions can be used as a formal way to manipulate a user’s priorities. Different strategies can be made optimal through changes of the payoff function. In ongoing work we are exploring under what conditions such “ideal payoff manipulations” can be made. We report some of our intermediate findings here.

Ideal payoff functions and manipulations

A payoff function is a function that translates *performance* on one or multiple tasks into a *single, explicit, objective currency* in a *consistent* manner. As a rule-of-thumb, the output of the function should be meaningful to the participant. Participants can then use the output of the payoff function to assess how well they are performing by comparing payoff values across trials and strategies. Similarly, a model can be used to compare payoff values of different strategies (e.g., Howes, et al., 2009; Janssen, et al., 2011). In this way, a *payoff curve* can be generated that captures how payoff fluctuates as a function of the possible strategies. An ideal payoff curve has four properties:

1. It has one global maximum.
2. It has no local maxima other than the global maximum.

3. The set of strategies that has a payoff value close to the maximum is narrow and these strategies are very similar in nature to the “optimum” strategy.
4. The distribution of possible payoff values is consistent and narrow, such that the mean payoff value of a strategy is representative of the distribution of values.

In essence, these properties guarantee that there is a unique, consistent, clear optimum strategy. This makes it easier for the participant and the modeler to identify the optimum strategy and to assess whether participants performed optimally. The black line in Figure 1 is an example of how payoff score (vertical axis) changes as a function of strategy (horizontal) in an ideal payoff curve.

If payoff is successful in manipulating performance, then in an ideal setting this can be used to make any arbitrary strategy “optimal” for at least one payoff function. In Figure 1 multiple alternative payoff curves are plotted in grey lines, as generated by hypothetical alternative payoff functions. This is considered an *ideal payoff manipulation*, because:

1. Each curve is an ideal payoff curve.
2. Across curves, each strategy is the optimum of at least one ideal payoff curve.

With this definition of an ideal payoff manipulation, we are currently exploring under what conditions (e.g., what types of tasks) such ideal manipulations are possible.

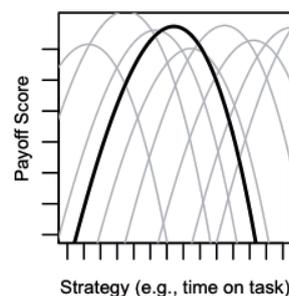


Figure 1: An ideal payoff manipulation (see text)

A mathematical model of interleaving

Inspired by our previous work on a tracking-while-typing task (Farmer, Janssen, & Brumby, 2011; Janssen, et al., 2011), we developed a mathematical model of this task. The model had to keep a one-dimensional first-order moving cursor inside a target area. The movement of the cursor was modeled using Pascal’s triangle, which can be used to give

exact predictions of the probabilities of the position of the cursor at each timestamp. The model could exert active control of the cursor to overwrite the random movement. In addition a simple secondary task was included, which solely involved opening a task window.

The model could only attend to one task at a time and experienced switch costs when switching between tasks. Both tasks were kept extremely simple on purpose, as this allowed us to focus on the role of payoff functions and whether an ideal payoff manipulation was possible, without having to worry about the correctness of each constraint and about the effects of constraints on performance.

Performance on both tasks was encapsulated in the payoff function. For the tracking task, the model gained points on every sample when the cursor was inside its target area; it lost points otherwise. For the secondary task, the model gained points whenever the window of this task was open; it lost points otherwise. The values of these four gain and loss components were systematically manipulated to explore whether ideal payoff manipulations were possible. We also explored the effects of using a log or exponential transformation to the functions. The general pattern of results was similar across these simulations.

Results and Discussion

As a first step to identify ideal payoff manipulations, we explored whether the location of the global maxima differed across payoff functions. In contrast to the definition of an ideal payoff manipulation, the strategy required to achieve the maximum payoff did not vary much as the payoff function varied. Only specific points emerged as maxima. These maxima were for strategies of which the performance of at least one of the underlying tasks (i.e., how many time units was the cursor inside the target area, how many time units was the secondary task window open) had a local maximum. These local maxima were themselves the result of the constraints imposed by the task environment (e.g., boundary of the tracking task target) and cognition (e.g., switch costs). That is, global maxima in the payoff curve emerged at positions where the interaction of the constraints led to beneficial performance trade-offs.

Looking at individual curves, many also violated the characteristics of an ideal payoff curve. Figure 2 shows three example curves. For each curve (different color lines), the strategy with the highest payoff is highlighted with an open circle. As can be seen, the curves violate the properties of an ideal payoff curve. There are local maxima and there are many strategies that achieved values close to the maximum value. This implies that fine tuned attentional strategies are not always required. We are therefore making further efforts to find task variants in which more subtle strategy choices are required.

Acknowledgments

This work was sponsored by EPSRC grant EP/G043507/1.

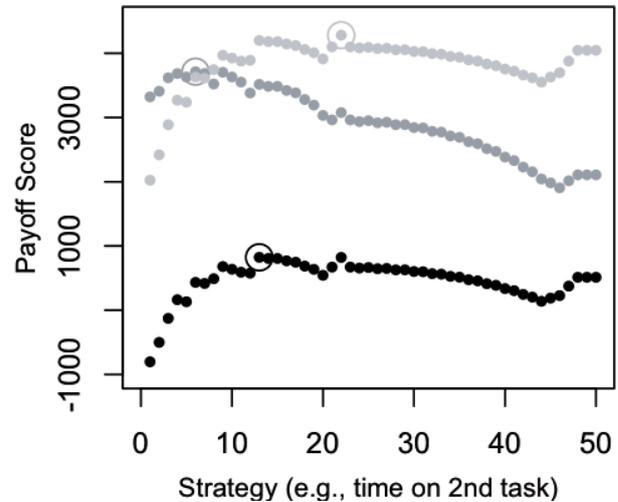


Figure 2: Three example payoff curves

References

- Farmer, G. D., Janssen, C. P., & Brumby, D. P. (2011). How long have I got? Making optimal visit durations in a dual-task setting. In *Proceedings of the 33rd Annual Meeting of the Cognitive Science Society* (pp. 862-867). Austin, TX: Cognitive Science Society.
- Howes, A., Lewis, R. L., & Vera, A. (2009). Rational adaptation under task and processing constraints: Implications for testing theories of cognition and action. *Psychological Review*, *116*, 717-751
- Janssen, C. P., Brumby, D. P., Dowell, J., Chater, N., & Howes, A. (2011). Identifying optimum performance trade-offs using a cognitively bounded rational analysis model of discretionary task interleaving. *Topics in Cognitive Science*, *3*, 123-139.
- Payne, S. J., Duggan, G. B., & Neth, H. (2007). Discretionary task interleaving: Heuristics for time allocation in cognitive foraging. *Journal of Experimental Psychology: General*, *136*, 370-388.
- Schumacher, E., Lauber, E., Glass, J., Zurbriggen, E., Gmeindl, L., Kieras, D. E., & Meyer, D. E. (1999). Concurrent response-selection processes in dual-task performance: Evidence for adaptive executive control of task scheduling. *Journal of Experimental Psychology: Human Perception and Performance*, *25*, 791-814.

What Can (and Can't) Make Problem Solving by Insight Possible? A Complexity-Theoretic Investigation

Todd Wareham (harold@mun.ca)

Department of Computer Science
Memorial University of Newfoundland
St. John's, NL Canada A1B 3X5

Keywords: Problem Solving; Insight; Representation Restructuring; (Parameterized) Computational Complexity

Problem Solving by Insight

Much of human problem solving can be accounted for by Newell and Simon's classic state-search model, in which a representation of a problem is chosen based on previous experience and search is then performed within the space of problem-states associated with this representation until a state is encountered that satisfies the problem's goals, *i.e.*, a solution. This assumes that the representation chosen is correct, in that there are solutions that can be reached by search within that representation's problem-state space. If this is not so, insights are necessary to modify the initial representation such that search can succeed (Duncker, 1945).

Problem solving by insight can be construed as cycles of search alternating with applications of special representation restructuring operators until a solution is reached. Within the most formally-stated such theory, the Representation Change Theory (RCT) of Knoblich et al. (1999), a problem representation additionally consists of a set of constraints encoding both restrictions on the search process and the characteristics of those problem-states that are solutions. The entities comprising a problem-state are grouped into chunks, where each chunk corresponds to a pattern that has proven useful in previous instances of problem solving. At any given time, only one set of chunks (whose members may not be nested) is considered active. RCT proposes two representation restructuring operations, namely, the removal of a particular constraint (Constraint Relaxation) or the replacement of an active chunk by its immediately-nested chunks (De-Chunking). The classical application by Knoblich et al. of RCT to matchstick arithmetic problems is shown in Figure 1.

Problem solving by insight is widely viewed as being more difficult than search-based problem solving (Chu & MacGregor, 2010). Within RCT, it has been conjectured that problems whose solutions violate smaller numbers of constraints (Knoblich et al., 1999, p. 1535) or with fewer constraints in total (MacGregor & Cunningham, 2009, p. 133) should be easier to solve. Though consistent with empirical observations, the question remains whether such explanations are complete, in the sense that the increases in solution-frequency and/or speed are due to these factors by themselves or by these factors in collaboration with other so-far unnoticed limitations. In this poster, we investigate these claims using the methodology for analyzing computational-level models of cognitive theories described in van Rooij & Wareham (2008).

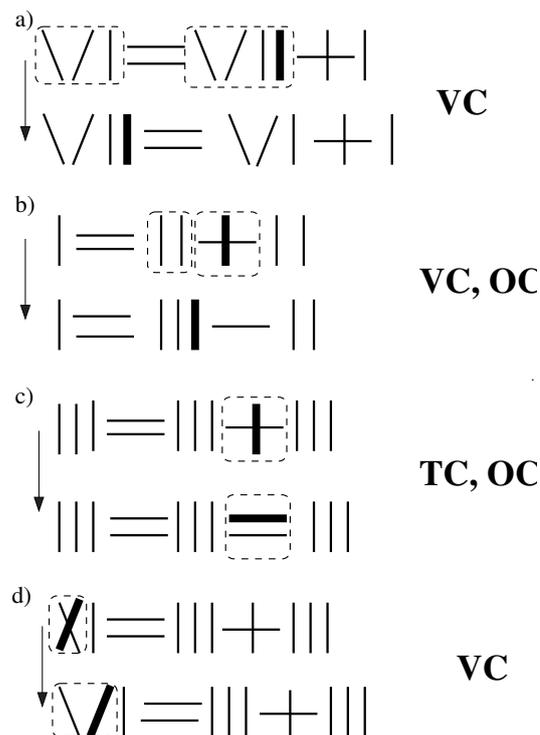


Figure 1: Solving Matchstick Arithmetic Problems within Representation Change Theory. In each problem, a single matchstick must be moved to make a mathematical formula correct. De-chunked chunks are denoted by dashed boxes. To the right of each problem are the constraints that were relaxed (VC: Value; OC: Operator; TC: Tautology).

Computational-level Model

A problem representation consists of a collection of entities and their relationships, a collection of chunks imposed on this collection, and a subset of those chunks comprising the currently active chunks. We model entity-relationship collections as predicate-structures, chunks as sub-predicate-structures, *i.e.*, a subset of the objects in a predicate-structure and all relationships in that structure that are based on the objects in this subset, and active chunks as non-nested collections of chunks that cover all objects (though not necessarily all predicates) in a predicate-structure. Given this, search operators are rules of the form $X \rightarrow Y$ that operate on predicate-structures, constraints are logic formulas which operate over

Table 1: Overview of parameters considered.

Name	Description
$ k_C $	Maximum # of constraint relaxations
$ k_D $	Maximum # of de-chunkings
$ k_S $	Maximum # of search operator applications
$ T $	Number of available chunk-types
$ C $	Number of constraints
$ O $	Number of available search operators
a	Maximum # of search-operator opportunities
$ T_A $	Maximum # of active chunks

the objects, predicates, and chunks in a problem representation. and the constraint relaxation and de-chunking operators are the deletion of a constraint and the replacement of an active chunk c by one or more non-overlapping chunks that are nested inside and cover all objects in c , respectively. This yields the following input-output mapping:

PROBLEM SOLVING BY RCT-INSIGHT (PSRI)

Input: Chunk-type set T , search-operator set O , problem representation p with active chunk-set D , constraint-set C , and integers k_C , k_D , and k_S .

Output: A solution s for p that is derived by applying $\leq k_C$ constraint relaxation and k_D de-chunking operators followed by $\leq k_S$ search operators from O , if such an s exists, and special symbol \perp otherwise.

Complexity Results and Discussion

Following convention in Computer (Garey & Johnson, 1979) and Cognitive (van Rooij, 2008) Science, we consider a cognitive theory tractable if its associated input-output mapping can be computed in polynomial time, *i.e.*, computed by an algorithm that runs in time upper-bounded by n^c where n is the input size and c is a constant.

Theorem 1 *PSRI is NP-hard when either $k_C = 0$ or $k_D = 0$.*

Theorem 1 establishes that, modulo the widely-believed conjecture that $P \neq NP$ (Fortnow, 2009), insight problem solving under RCT cannot be done in polynomial time. Note that this holds whether re-structuring consists purely of constraint relaxation or de-chunking, which implies that the focus to date on restricting only the amount of constraint relaxation to ease the difficulty of solving insight problems is in error.

This result also means that strong restrictions must be assumed to apply to the input domain of PSRI for RCT to be able to explain solution of insight problems by human beings. Let us formulate such restrictions in terms of the values of selected parameters, which are aspects of problem inputs. We say that a set K of one or more parameters renders an input-output mapping Π **fixed-parameter (fp) tractable** if there is an algorithm for Π that runs in time upper-bounded by $f(K)n^c$, where f is an arbitrary function (Downey & Fellows, 1999; van Rooij, 2008). To investigate which restric-

tions suffice for rendering PSRI tractable, we performed fp-tractability analyses relative to the parameters in Table 1.

Theorem 2 *PSRI is not fp-tractable for $\{k_C, k_D, k_S, |T|, |C|, |O|, a\}$,*

Theorem 3 *PSRI is fp-tractable for $\{k_D, k_S, |C|, a, |T_A|\}$,*

Theorem 2 establishes that PSRI cannot be made easy even if the parameters in both published conjectures (k_C and $|C|$) are bounded simultaneously with a number of other plausible parameters. Theorem 3 provides the first provably complete explanation of the tractability of PSRI. This explanation is not totally satisfactory because (1) there is no empirical evidence that $|C|$ and a are small in practice and (2) the invoked parameter-set is not provably minimal, as it is possible that restricting some subset of these parameters in combination with $|T_A|$ may give fp-tractability. That being said, whether or not either of these objections are substantive can be settled by further experimental and theoretical research. Future research should also investigate whether additional parameters not considered here yield alternate complete explanations of the precise circumstances under which problem solving by insight is and is not possible, both under RCT and other proposed theories of restructuring-assisted problem solving (see Ash et al. (2009) and references).

References

- Ash, I. K., Cushen, P. J., & Wiley, J. (2009). Obstacles in investigating the role of restructuring in insightful problem solving. *Journal of Problem Solving*, 2(2), 6–41.
- Chu, Y., & MacGregor, J. N. (2010). Human performance on insight problem solving. *Journal of Problem Solving*, 3(2), 119–150.
- Downey, R., & Fellows, M. (1999). *Parameterized Complexity*. Berlin: Springer.
- Duncker, K. (1945). *On problem solving* (Vol. 58; L. S. Lees, Trans.) (No. 5). American Psychological Association.
- Fortnow, L. (2009). The Status of the P Versus NP Problem. *Communications of the ACM*, 52(9), 78–86.
- Garey, M. R., & Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: W.H. Freeman.
- Knoblich, G., Ohlsson, S., Haider, H., & Rhenius, D. (1999). Constraint relaxation and chunk decomposition in insight problem solving. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 25(6), 1534–1555.
- MacGregor, J. N., & Cunningham, J. B. (2009). The effects of number and level of restructuring in insight problem solving. *Journal of Problem Solving*, 3(2), 130–141.
- van Rooij, I. (2008). The Tractable Cognition Thesis. *Cognitive Science*(32), 939–984.
- van Rooij, I., & Wareham, T. (2008). Parameterized Complexity in Cognitive Modeling: Foundations, Applications and Opportunities. *Computer Journal*, 51(3), 385–404.

A Concept of a Cognitive System of Motivation (COSMO)

Andreas Ammann (andreas.ammann@biodidaktik.uni-halle.de)

Department of Didactics of Biology, Biologicum, Weinbergweg 10
Halle / S., 06120 Germany

Keywords: cognitive system; motivation; COSMO; goal striving; mediating states; optimization

Introduction

Motivational concepts, such as approach and avoidance goals, have an increasing impact on cognitive models (Markman et al., 2005). In this paper a concept of a cognitive system of motivation (COSMO) is presented that focuses on goal striving by calculating an optimal investment of resources. As a basis a mathematical model of interaction between agents and environments was developed. The intention is to design a general cognitive system that can be used for a wide range of environments serving as an operator for more specific cognitive systems. A second aim is to predict motivational mechanisms by concentrating on characteristics of biological agents.

General mathematical model of interaction

Goals can be arranged as complex goal systems where subgoals are set to pursue the main goals (Kruglanski, 2002). As biological agents act in real environments the interaction between environments and agent is essential for effective goal setting. The diversity of environmental components is a challenge for a cognitive system as it must handle a spectrum of different components and tasks. Evolutionary psychologists argue that the ability to efficiently handle diverse tasks is strictly limited by evolutionary history (Smith et al., 2001). In the presented approach requirements for cognitive systems are reduced by focusing on general characteristics of diverse components rather than their differences: states of environmental components are characterized by numerical values, agents are able to change these state by investing resources and environments are highly cross-linked systems. A general mathematical model of interaction between an agent and environments was constructed. Environmental components are merged into fields. Resource investment of an agent changes the numeric value of a field and due to cross-linking also the total network of fields. The internal information about these states is considered as mediating states (Markman & Dietrich, 2000). In this model the action of an agent is defined as the investment of resources into a field.

Optimizing investment of resources

In the presented cognitive system the superior goal is the maximization of the numerical value of a primary field by an optimal investment of resources into other fields (Fig. 1).

Other more specific cognitive systems can be interpreted as fields so that these systems can be used for optimization by allocating resources to them. Because biological agents have only limited information about complex environments the focus of this cognitive system is on optimization approaches with a rather low demand for precise information. Following the paradigm “motivation as cognition” (Kruglanski, 2002) this action controlling system can be considered as a cognitive system of motivation (COSMO), optimization calculations as motivational mechanisms and the results as motivation. By doing so components of COSMO show high similarities to a wide range of concepts of motivation psychology.

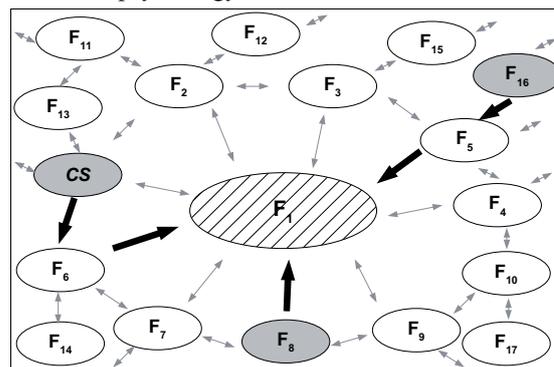


Figure 1: Working principle of COSMO. For maximization of the value of the primary field (F_1) COSMO allocates resources into cross-linked fields (grey ovals) or other cognitive systems (CS). Strong cross-links are marked as black arrows, weak ones as grey arrows.

References

- Kruglanski, A. W., Sha, J. Y., Fishbach, A., Friedman, R., Chun, W. Y. & Sleeth-Keppler, D. (2002). A theory of goal systems. In M. P. Zanna (Ed.), *Advances in Experimental Social Psychology*. San Diego: Academic Press.
- Markman, A. B. & Dietrich, E. (2000). In defence of representation. *Cognitive Psychology*, 40, 138-171.
- Markman, A. B., Maddox, W. T. & Baldwin, G. C. (2005). The implication of advances on motivation for cognitive models. *Journal of Experimental & Theoretical Artificial Intelligence*, 17, 371-384.
- Smith, E. A., Mulder, M. B. & Hill, K. (2001). Controversies in the evolutionary social sciences: a guide for the perplexes. *TRENDS in Ecology & Evolution*, 16, 128-135.

A Computational Model for Forms of Selective Attention based on Cognitive and Affective Feelings

Luis Macedo (macedo@dei.uc.pt)

University of Coimbra

Centre for Informatics and Systems of the University of Coimbra, Coimbra, Portugal

Abstract

Inspired on natural selective attention studies, we propose a computational model of selective attention that relies on the assumption that uncertain, surprising and motive congruent/incongruent information demands attention from an intelligent agent. This computational model has been integrated into the architecture of a Belief-Desire-Intention artificial agent so that this can autonomously select relevant, interesting information of the (external or internal) environment while ignoring other less relevant information. The advantage is that the agent can communicate only that interesting, selective information to its processing resources (focus of the senses, decision-making, etc.) or to its human owner's processing resources so that these resources can be allocated more effectively. We illustrate and provide experimental results of this role of the artificial, selective attention mechanism in the time-critical, risky situation, of driving a vehicle, by showing that it prevents both the personal traffic assistant agent's and its human owner's decision-making resources of receiving unnecessary traffic information.

Keywords: Selective attention; Interest; Value of information, Surprise; Uncertainty; Information overload; Resource-bounded agents; Personal agents.

Introduction

In many ways, the advent of information technology is a primary reason for the abundance of information with which humans are inundated, due to its ability to produce more information more quickly and to disseminate this information to a wider audience than ever before. Contrary to what in general could be expected, a lot of recent studies confirmed what Alvin Toffler (1970) predicted a few decades ago: the overabundance of information instead of being beneficial is a huge problem having many negative implications not only in personal life but also in organizations, business, and in general in the world economy. Research proves that the brain simply does not deal very well with this multitasking process: there is a waste of time as the brain switches from one task to another and back again (Klingberg, 2008). This explains why decision quality and the rate of performing tasks degrades with increases in the amount of information being considered.

A fundamental strategy for dealing with this problem of information overload (O'Connell, 2008) should include making devices that incorporate themselves selective attention agents in order to decrease the amount of information considered in their own reasoning/decision-making processes or decrease the amount of information provided by them to humans, preventing these from a number of interruptions.

But how to model selective attention in artificial agents? The problem starts at the human level. Although selective attention has been thoroughly researched over the last 100 years in psychology and more recently in neuroscience (e.g.,

Kahneman, 1973; Wright & Ward, 2008), at present there is no general theory of selective attention. Instead there are specific theories for specific tasks such as orienting, visual search, filtering, multiple action monitoring (dual task), and multiple object tracking.

In spite of this, a number of models of selective attention has been proposed in Cognitive Science (e.g., Horvitz, Jacobs, & Hovel, 1999). Particularly related with these models is the issue of measuring the value of information. A considerable amount of literature has been published on these measures, especially from the fields of active learning and experimental design. Most of those measures rely on assessing the utility or the informativeness of information (e.g., Horvitz & Barry, 1995; MacKay, 1992; Lindley, 1955; Settles, 2008). However, little attention has been given to the surprising and motive congruence value of information, giving the beliefs and desires of an agent.

Opposed to other approaches (e.g., Itti & Baldi, 2006; Peters, 1998; Schmidhuber, 2006; Oudeyer, Kaplan, & Hafner, 2007) relying on low-level, raw information, Macedo, Cardoso, and Reizenzein (2001; 2004), and Lorini and Castelfranchi (2007) proposed, independently, computational models of surprise that are based on the mechanism that compares newly acquired beliefs to preexisting beliefs. Both models of artificial surprise were influenced by psychological theories of surprise (e.g., Meyer, Reizenzein, & Schützwohl, 1997), and both seek to capture essential aspects of human surprise (see Macedo, Cardoso, Reizenzein, Lorini, & Castelfranchi, 2009, for a comparison). In agreement with most theories of human surprise, both models of artificial surprise conceptualize surprise as a fundamentally expectation- or belief-based cognitive phenomenon, that is, as a reaction to the disconfirmation of expectations or, more generally, beliefs. Furthermore, in both models, beliefs are understood as propositional attitudes (e.g., Searle, 1983), and a quantitative belief concept (subjective probability) is used. Both artificial surprise models draw a distinction between two main kinds of expectations or beliefs whose disconfirmation causes surprise (see also Ortony & Partridge, 1987): Active versus passive expectations. Although Macedo and Cardoso initially used the same surprise intensity function, according to which the intensity of surprise about an event is proportional to its unexpectedness, Macedo, Reizenzein and Cardoso subsequently opted for a "contrast model" of surprise intensity. This model assumes that the intensity of surprise about an event reflects its probability difference to the contextually most expected event (see also Teigen & Keren, 2003).

In this paper we describe an artificial selective attention mechanism that may be used by artificial agents so that only cognitively and affectively, interesting/relevant information is selected and forwarded to reasoning/decision-making units. Our approach relies on the psychological and neuroscience studies about selective attention which defend that variables such as unexpectedness, unpredictability, surprise, uncertainty, and motive congruence demand attention (e.g., Berlyne, 1960; Kahneman, 1973; Ortony & Partridge, 1987). One of the features of the selective attention mechanism is that it should work in the absence of a model of decision-making of the artificial agent, or of its designer, owner or user for whom the artificial agent might act on his/her behalf.

The next section describes the computational model of selective attention, focusing on how the multidimensional value of information is computed, which will be illustrated with an example in Section 3. Section 4 examines the performance of the selective attention mechanism as well as its role on the decrease of unnecessary information while not affecting significantly an agent decision-making performance. Finally, in Section 5 we present conclusions.

A Computational Model for Forms of Selective Attention

Selective attention may be defined as the cognitive process of selective allocation of processing resources (focus of the senses, etc.) on relevant, important or interesting information of the (external or internal) environment while ignoring other less relevant information. The issue is how to measure the value of information. What makes something interesting? In cognitive science, attentional focus is linked with expectation generation and failure, i.e., with surprise (Ortony & Partridge, 1987). Therefore, it is reasonable to consider that any model of selective attention should rely on a cognitive model of surprise. However, surprise is not enough. Happiness/pleasantness, which according to cognitive theories of emotion and specifically to belief-desire theories of emotion (Reisenzein, 2008) is directly related to congruence between new information and the human agent’s motives/desires, may also play also a fundamental role on attention. For this reason, the system must also incorporate a measure of the expected satiation of the desires.

In order to accomplish all those requirements, we developed an architecture for a personalized, artificial selective attention agent (see Figure 1). We assume: (i) this agent interacts with the external world receiving from it information through the senses and outputs actions through their effectors; (ii) the world is described by a large amount of statistical experiments; (iii) the agent is a BDI agent (Rao & Georgeff, 1995), exhibiting a prediction model (model for generating expectations, i.e., beliefs about the environment), a desire strength prediction model (a model for generating desire strengths for all the outcomes of the statistical experiments of the world that are known given the desires of the agent – profile of the agent which include basic desires), as well as

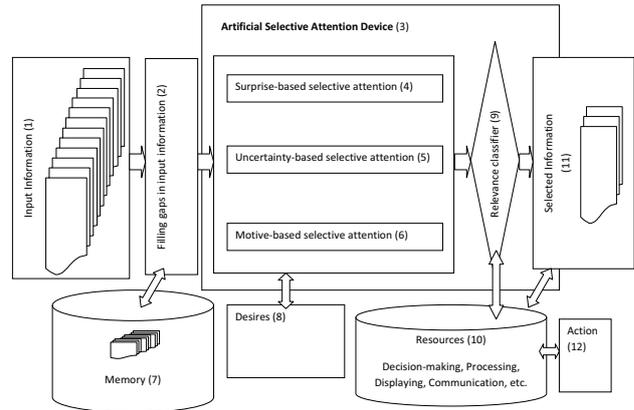


Figure 1: Architecture of an artificial selective attention agent.

the intentions (these define the profile of the agent); (iv) the agent contains other resources for the purpose of reasoning and decision-making.

While the belief strengths are inferred from data using a frequentist approach and updated as new information is acquired, the desirability of the outcomes is previously set up although they depend on the intention of the agent, suffering changes whenever the agent is committed with a new intention.

The first of the modules of the architecture (module 1 in Figure 1) is concerned with getting the input information. The second is the computation of the current world state. This is performed by generating expectations or assumptions (module 2), based on the knowledge stored in memory, for the gaps of the environment information provided by the sensors (module 1). We assume that each piece of information resulting from this process, before it is processed by other cognitive skills, goes through several sub-selective attention devices, each one evaluating information according to a certain dimension such as surprise (module 4), uncertainty (module 5), and motive-congruence/incongruence – happiness (module 6). For this task the selective attention mechanism takes into account some knowledge container (memory — pre-existing information (module 7)), and the intentions and desires (motives — module 8). There is a decision-making module (module 9) that takes into account the values computed by those sub-selective attention modules and decides if a piece of information is relevant/interesting or not. Then, this module of decision-making selects the higher relevant pieces of information so that other resources (reasoning, decision-making, displaying, communication resources, etc.) (module 10) can be allocated to deal with them.

The representation of the memory contents (beliefs) relies on semantic features or attributes much like in semantic networks (Russell & Norvig, 2010) or schemas (Rumelhard & Ortony, 1977). Each attribute, $attr_i$, viewed by us as a statistical experiment, is described by a probabilistic distri-

bution, i.e., a set $A_i = \{ \langle value_j, prob_j, desireStrength_j \rangle : j = 1, 2, \dots, n \}$, where n is the number of possible values of the attribute, $P(attr_i = value_j) = prob_j$, and $desireStrength_j$ is the desirability of $attr_i = value_j$ (for a related work see (Reisenzein, 2008)).

The next sub-sections describe each one of the dimensions for evaluating information, namely surprise, uncertainty, and motive congruence/incongruence. While the dimensions of surprise and uncertainty are related to the value of information to the belief store of the agent, the dimension of motive congruence/incongruence is related to the value of information to the goals/desires of the agent (these dimensions are related to the concepts of cognitive and affective feelings of (Clore, 1992) and belief-belief and belief-desire comparators of (Reisenzein, 2008)).

Surprise Value of Information

We adopted the computational model of surprise of (Macedo & Cardoso, 2001; Macedo et al., 2004) which is formally defined in Definition 1 (for related models see Macedo et al., 2009). Macedo, Cardoso and Reisenzein computational model of surprise suggests that the intensity of surprise about an event E_g , from a set of mutually exclusive events E_1, E_2, \dots, E_m , is a nonlinear function of the difference, or contrast, between its probability and the probability of the highest expected event E_h in the set of mutually exclusive events E_1, E_2, \dots, E_m .

Definition 1 Let (Ω, A, P) be a probability space where Ω is the sample space (i.e., the set of possible outcomes of the experiment), $A = A_1, A_2, \dots, A_n$ is a σ -field of subsets of Ω (also called the event space, i.e., all the possible events), and P is a probability measure which assigns a real number $P(F)$ to every member F of the σ -field A . Let $E = \{E_1, E_2, \dots, E_m\}$, $E_i \in A$, be a set of mutually exclusive events in that probability space with probabilities $P(E_i) \geq 0$, such that $\sum_{i=1}^m P(E_i) = 1$. Let E_h be the highest expected event from E . The intensity of surprise about an event E_g from E is given by:

$$S(E_g) = \log(1 + P(E_h) - P(E_g)) \quad (1)$$

The probability difference between $P(E_h)$ and $P(E_g)$ can be interpreted as the amount by which the probability of E_g would have to be increased for E_g to become unsurprising.

Proposition 1 In each set of mutually exclusive events, there is always at least one event whose occurrence is unsurprising, namely, E_h .

Uncertainty-based Value of Information

Information is a decrease in uncertainty which, according to information theory, is measured by entropy (Shannon, 1948). When new information is acquired its amount may be measured by the difference between the prior uncertainty and the posterior uncertainty.

Definition 2 Let (Ω, A, P_{prior}) be a probability space where Ω is the sample space (i.e., the set of possible outcomes of the experiment), $A = A_1, A_2, \dots, A_m$ is a σ -field of subsets of Ω (also called the event space, i.e., all the possible events), and P_{prior} is a probability measure which assigns a real number $P_{prior}(F)$ to every member F of the σ -field A . Let $E = \{E_1, E_2, \dots, E_m\}$, $E_i \in A$, be a set of mutually exclusive events in that probability space with probabilities $P_{prior}(E_i) \geq 0$, such that $\sum_{i=1}^m P_{prior}(E_i) = 1$. Let P_{post} be the posterior probability measure, after some data is acquired, which assigns a real number $P_{post}(F)$ to every member F of the σ -field A such that it assigns $P_{post}(E_i) \geq 0$ with $\sum_{i=1}^m P_{post}(E_i) = 1$. According to information theory, the information gain of an agent after some data is acquired, $IG(E)$, is given by the decrease in uncertainty:

$$\begin{aligned} IG(E) &= H_{prior}(E) - H_{post}(E) \\ &= - \sum_{i=1}^m P_{prior}(E_i) \times \log(P_{prior}(E_i)) - \\ &\quad \left(- \sum_{i=1}^m P_{post}(E_i) \times \log(P_{post}(E_i)) \right) \quad (2) \end{aligned}$$

$H_{post} = 0$ if and only if all the $P_{post}(E_i)$ but one are zero, this one having the value unity. Thus only when we are certain of the outcome does H_{post} vanish, otherwise it is positive.

IG is not normalized. In order to normalize it we must divide it by $\log(m)$ since it can be proved that $IG \leq \log(m)$:

$$IG(E) = \frac{H_{prior}(E) - H_{post}(E)}{\log(m)} \quad (3)$$

Motive Congruence/Incongruence-based Value of Information

While the measure of surprise takes into account beliefs that can be confirmed or not, the pleasantness function that we describe in this subsection takes as input desires that, contrary to beliefs, can be satisfied or frustrated. Following the belief-desire theory of emotion (Reisenzein, 2008), we assume that an agent feels happiness if it desires a state of affairs (a proposition) and firmly believes that that state of affairs obtains. The intensity of happiness about an event is a monotonically increasing function of the degree of desire of that event as formally defined in Definition 4.

Definition 3 Let (Ω, A) be a measurable space where Ω is the sample space (i.e., the set of possible outcomes of the experiment) and $A = A_1, A_2, \dots, A_m$ a σ -field of subsets of Ω (also called the event space, i.e., all the possible events). We define the measure of desirability of an event on (Ω, A) as $D : A \rightarrow [-1, 1]$, i.e., as a signed measure which assigns a real number $-1 \leq D(F) \leq 1$ to every member F of the σ -field A based on the profile of the agent, so that the following properties are satisfied:

- $D(\emptyset) = 0$, $-|\Omega| \geq D(\Omega) \leq |\Omega|$
- if A_1, A_2, \dots is a collection of disjoint members of A , in that $A_i \cap A_j = \emptyset$ for all $i \neq j$, then

$$D\left(\bigcup_{i=0}^{\infty} A_i\right) = \sum_{i=0}^{\infty} D(A_i) \quad (4)$$

The triple (Ω, A, D) is called the desirability space.

Definition 4 Let (Ω, A, P) and (Ω, A, D) be the probability and the desirability spaces described, respectively, in Definition 1 and Definition 3. Let $E = \{E_1, E_2, \dots, E_m\}$, $E_i \in A$, be a set of mutually exclusive events in that probability space with probabilities $P(E_i) \geq 0$, $\sum_{i=1}^m P(E_i) = 1$. If $P(E_g) = 1$, the intensity of happiness, i.e., motive congruence, about an event E_g from E is given by:

$$MC(E_g) = D(E_g) \quad (5)$$

The Principle of Selective Attention

Having defined the motive, the uncertainty-based, and surprise-based selective attention modules, we are now in a position to formulate, in a restricted sense (without the inclusion of other information measures such as complexity), the principle that a resource-bounded rational agent should follow in order to avoid an overabundance of information and interruptions in the absence of a model for decision-making. Note that if this model is known, the problem is reduced to the classical computation of the value of information that has been extensively studied (e.g., Horvitz & Barry, 1995; Russell & Norvig, 2010).

Definition 5 A resource-bounded rational agent should focus its attention only on the relevant and interesting information, i.e., on information that is congruent or incongruent to its motives/desires, and that is cognitively relevant because it is surprising or because it decreases uncertainty.

We may define real numbers α , β , and γ as levels above which the absolute values of motive congruency, surprise, and information gain (decrease of uncertainty), respectively, should be so that the information can be considered valuable or interesting. These are what we called the triggering levels of alert of the selective attention mechanism. Note that, making one of those parameters null is equivalent to removing the contribution of the corresponding component from the selective attention mechanism.

Practical Application

Advanced Travel Information Systems (ATIS) are designed to assist travelers in making pre-trip and en-route travel decisions by providing them pre-trip and en-route information. However, while these information systems can undoubtedly help humans perform better in these complex traveling scenarios, they might provide an unhandled amount of information to humans that may compromise their performance.

It is contended that while many traveler information systems are innovative and make use of cutting edge technologies, they lack real machine intelligence and therefore may be limited in their ability to service the traveling public over the long-run. On the one hand, a wave of technological developments, in particular the increasing deployment of GIS and, on the other hand, the introduction and rapid market penetration of mobile devices such as cell phones boosted the development of ATIS towards what has been termed Intelligent Traveler Information Systems (ITIS) (Adler & Blue, 1998), in which artificial intelligence techniques are drawn upon to create systems capable of providing travelers with more personalized planning assistance. This is the goal of integrating selective agents in personal devices that receive information from the ATIS to act as personal assistant selective attention agents in order to avoid unnecessary interruptions to their users by enabling that only interesting information (i.e., with a value above a threshold defined by the user) is provided to them.

We are developing an ITIS that receives information about the traffic conditions and sends it to the mobile devices of the travelers. All that collected information is stored in the knowledge base/memory of the ITIS. There is a personal selective attention agent for each registered traveler. Each one of these personal agents has information about the expectations of its owner based on their travel history.

Let us illustrate how the value of information is computed by the selective attention mechanism. Suppose that a traveller's navigation system provided the pre-route path containing a road A for an agent (a driver) based on its profile (e.g., preference for shortest routes). Suppose the agent has the following expectations for the traffic conditions of road A , for a certain period/time of the day for a certain day of the week: 60% of probability of "good traffic conditions" (event E_1), 30% of probability of "moderate traffic conditions" (event E_2), and 10% of probability of "bad traffic conditions" (event E_3). Suppose the desire strengths of these events are 1, -0.5, and -1, respectively. Given that the agent plans to go through that route, suppose its module for generating/managing desires assigns a null desire strength for the other routes as it does not care about the traffic conditions of the other roads that are not part of its planned route. What is the relevance of becoming aware that the current traffic conditions of road A are good (event E_1)? Considering solely the motive-based component, the outcomes (events E_1 , E_2 , and E_3) elicits happiness (motive congruence) with intensity 1, -0.5 and -1, respectively. E_1 is congruent/consistent with the goals of the agent, while E_2 and E_3 are incongruent with the goals of the agent.

According to Equation 1, the surprise value of E_1 , E_2 , and E_3 are, respectively, 0, 0.38, and 0.58. Illustrating for the case of E_3 :

$$\begin{aligned} \text{Surprise}(E_3) &= \log(1 + P(E_1) - P(E_3)) \\ &= \log(1 + 0.6 - 0.1) = 0.58 \end{aligned} \quad (6)$$

According to Equation 3, the normalized information gain value of E_1 , E_2 , or E_3 is:

$$\begin{aligned} IG(E) &= \frac{H_{prior}(E) - H_{post}(E)}{\log(m)} = \frac{H_{prior}(E) - 0}{\log(3)} \\ &= \frac{-\sum_{i=1}^3 P_{prior}(E_i) \times \log(P_{prior}(E_i))}{\log(3)} = 0.82(7) \end{aligned}$$

Assume the Principle of Selective Attention described above, with parameters $\alpha = 0.3$, $\beta = 0.5$, and $\gamma = 0.6$. Are all these events interesting? Considering the motive-based component all those events are interesting. However, from the perspective of the surprise-based selective attention component, the answer is "no" to the question related with the events E_1 and E_2 in that their surprise values, 0 and 0.38, respectively, are below β . With respect to E_3 the answer is "yes" given that its surprise value is 0.58. Taking the uncertainty-based component into account, the answer is "yes" for all the events because their occurrence gives a normalized information gain of 0.82 which is above γ .

Experiment

We conducted an experiment to evaluate the performance and the potential benefits of the personal selective attention model for filtering unnecessary information for human travelers. To do that we assessed its performance considering the opinions of the human travelers, comparing their classifications about whether some information is relevant or not and the classifications of the selective attention agent. The selective attention agent is considered to perform erroneously if it filters a relevant information or if it does not filter an irrelevant information. The environment considered was Bissaya Barreto Avenue of the city of Coimbra, Portugal. We configured a selective attention agent to provide real time information about the traffic conditions in that street to 5 volunteer travelers whose path include that street. We collect information about the relevance of the information the agent delivered during 10 days at the same time (9h:00m) and always concerning the same street. In addition, after the trip, the information the agent didn't delivered, when the value computed by its selective attention mechanism was below the triggering level of alert, was shown to the travelers and these were asked to rate the relevance they would had assigned that information if it was delivered. All these data were used to compute the true and false positives. In each situation, the human agent, if prevented from receiving information, maintains the plan suggested by the navigation system, otherwise, if informed, he/she may consider alternative routes and change its mind by planning to proceed through one of those alternative route.

The parameters considered were $\alpha = 0.3$, $\beta = 0.3$, $\gamma = 0.6$. These are average values obtained from a questionnaire presented to human drivers in which they were asked to specify reasonable values for those parameters. Table 1 presents the confusion matrix for this model. We found evidence indicating that the selective attention mechanism (with those

Table 1: Confusion matrix of the selective attention mechanism for $\alpha = 0.3$, $\beta = 0.3$, and $\gamma = 0.6$.

	Prevented	Not Prevented	
Not relevant	64.55%	9.09%	73.64%
Relevant	19.09%	7.27%	26.36%
	83.64%	16.36%	100%

parameters) contributes significantly to decrease the amount of irrelevant information by an average of 83.64% ($p=0.000$). Furthermore, and not less important, we found evidence indicating that, preventing the human agent from receiving that amount of irrelevant information, the performance of the agent was not affected significantly. In fact, we found that from these there is only an average of 22.83% (corresponding to 19.09% of all the information) of false negatives, which indicates those means are statistically different ($p=0.000$). However, with respect to the false positives, we found that from the 16.36% of interruptions, an average of 55.56% (corresponding to 9.09% of all the information) was not relevant, which are not significantly different ($p=0.146$) and, therefore, we can not reject the null hypothesis in this case.

The accuracy of this specific model is 0.72, while the recall (true positive rate) and the precision are 0.28 and 0.44, respectively. The F1 measure is 0.34.

Discussion and Conclusions

We presented a computational model for selective attention based on cognitive and affective feelings. We found evidence indicating that the mechanism contributes for decreasing the amount of unnecessary information while maintaining acceptable the performance of the owner (a human).

The advantages of reasoning correctly with less information include spending less time in processing information which is important in time-critical, high-risk situations. Besides, agents equipped with a selective attention filter can be successful personal assistants of humans, integrated for instance in mobile devices, so that their human users are prevented from unnecessary interruptions. This may be of high value in critical situations such as driving a car in that, as reported by (Horvitz & Barry, 1995), numerous cognitive studies have provided evidence of the problems in information processing exhibited by humans when dealing with large amounts of information such as that the speed at which humans perform tasks drops as the quantity of information being considered increases, and that the rate of performing tasks can be increased by filtering irrelevant information. In this particular case of transportation systems, the ultimate advantage may be less vehicle accidents and less deaths, while in organizations the advantage may be an improvement in their workers productivity and therefore less costs.

An hypothesis that might be risen is that taking other sub-selective attention modules such as those based on other cognitive or affective feelings (Clore, 1992) (e.g., familiarity, complexity) into account improves the performance of the

mechanism. More experiments should be done with this aim.

Acknowledgments

Work funded by Fundação para a Ciência e Tecnologia — Project PTDC/EIA-EIA/108675/2008, and by FEDER through Programa Operacional Factores de Competitividade do QREN — COMPETE:FCOMP-01-0124-FEDER-010146.

References

- Adler, J., & Blue, V. (1998). Toward the design of intelligent traveler information systems. *Transportation Research Journal Part C*, 6/3, 157-172.
- Berlyne, D. (1960). *Conflict, arousal and curiosity*. New York: McGraw-Hill.
- Clore, G. (1992). Cognitive phenomenology: Feelings and the construction of judgment. In L. Martin & A. Tesser (Eds.), *The construction of social judgments* (p. 133-163). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Horvitz, E., & Barry, M. (1995). Display of information for time-critical decision-making. In *Proceedings of the eleventh conference on uncertainty in artificial intelligence* (p. 296-305). Morgan Kaufmann.
- Horvitz, E., Jacobs, A., & Hovel, D. (1999). Attention-sensitive alerting. In *Proceedings of the fifteenth conference on uncertainty and artificial intelligence* (p. 305-313). Morgan Kaufmann.
- Itti, L., & Baldi, P. (2006). Bayesian surprise attracts human attention. *Advances in Neural Information Processing Systems (NIPS 2005)*, 19, 1-8.
- Kahneman, D. (1973). *Attention and effort*. Englewood Cliffs, NJ: Prentice-Hall.
- Klingberg, T. (2008). *The overflowing brain: information overload and the limits of working memory*. Oxford University Press.
- Lindley, D. (1955). On a measure of information provided by an experiment. *The Annals of Mathematical Statistics*, 986-1005.
- Lorini, E., & Castelfranchi, C. (2007). The cognitive structure of surprise: looking for basic principles. *Topoi: An International Review of Philosophy*, 26(1), 133-149.
- Macedo, L., & Cardoso, A. (2001). Modelling forms of surprise in an artificial agent. In J. Moore & K. Stenning (Eds.), *Proceedings of the 23rd annual conference of the cognitive science society* (p. 588-593). Edinburgh, Scotland, UK: Erlbaum.
- Macedo, L., Cardoso, A., Reizenzein, R., Lorini, E., & Castelfranchi, C. (2009). Artificial surprise. In J. Vallverd & D. Casacuberta (Eds.), *Handbook of research on synthetic emotions and sociable robotics: New applications in affective computing and artificial intelligence* (p. 267-291). Hershey: USA: IGI Global.
- Macedo, L., Reizenzein, R., & Cardoso, A. (2004). Modeling forms of surprise in artificial agents: empirical and theoretical study of surprise functions. In K. Forbus, D. Gentner, & T. Regier (Eds.), *Proceedings of the 26th annual conference of the cognitive science society* (pp. 873-878). Chicago, Illinois, USA: Lawrence Erlbaum Associates, Inc.
- MacKay, D. (1992). Information-based objective functions for active data selection. *Neural Computation*, 4(4), 590-604.
- Meyer, W. . U., Reizenzein, R., & Schützwohl, A. (1997). Towards a process analysis of emotions: The case of surprise. *Motivation and Emotion*, 21, 251-274.
- O'Connell, N. (2008). Interruption overload. *Strategic Direction*, 24(10), 3-5.
- Ortony, A., & Partridge, D. (1987). Surprisingness and expectation failure: what's the difference? In *Proceedings of the 10th international joint conference on artificial intelligence* (p. 106-8). Milan, Italy: Morgan Kaufmann.
- Oudeyer, P., Kaplan, F., & Hafner, V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, 11(2), 265-286.
- Peters, M. (1998). Towards artificial forms of intelligence, creativity, and surprise. In *Proceedings of the twentieth annual conference of the cognitive science society* (p. 836-841). Madison, Wisconsin, USA: Erlbaum.
- Rao, A., & Georgeff, M. (1995). Bdi agents: from theory to practice. In *Proceedings of the first international conference on multiagent systems* (p. 312-319). San Francisco, CA, USA: MIT Press.
- Reizenzein, R. (2008). Emotions as metarepresentational states of mind: Naturalizing the belief-desire theory of emotion. *Cognitive Systems Research*, 9.
- Rumelhardt, D., & Ortony, A. (1977). The representation of knowledge in memory. In R. Anderson, R. Spiro, & W. Montague (Eds.), *Schooling and the acquisition of knowledge* (p. 99-135). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Russell, S., & Norvig, P. (2010). *Artificial intelligence - a modern approach* (3rd ed.). Englewood Cliffs, NJ: Prentice Hall.
- Schmidhuber, J. (2006). Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. *Connection Science*, 18, 173-187.
- Searle, J. (1983). *Intentionality*. Cambridge: Cambridge University Press.
- Settles, B. (2008). *Curious machines: Active learning with structured instances*. Unpublished doctoral dissertation, University of Wisconsin, Madison, USA.
- Shannon, C. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27, 379-423 and 623-656.
- Teigen, K. H., & Keren, G. B. (2003). Surprises: Low probabilities or high contrasts? *Cognition*, 87, 55-71.
- Toffler, A. (1970). *Future shock*. Bantam Book.
- Wright, R. D., & Ward, L. M. (2008). *Orienting of attention*. Oxford, UK: Oxford University Press.

A process model of immediate inferences

Sangeet Khemlani¹, J. Gregory Trafton¹, Max Lotstein², and P. N. Johnson-Laird²
khemlani@aic.nrl.navy.mil, trafton@itd.nrl.navy.mil, lotstein@princeton.edu, phil@princeton.edu

¹NCARAI, Code 5515, Naval Research Laboratory, Washington, DC 20375 USA

²Department of Psychology, Princeton University, Princeton, NJ 08540 USA

Abstract

Individuals can make inferences from a single quantified premise. For instance, if you know that *all of the Virginians are students*, you might infer that *some of the students are Virginians*. We describe a computational system, *mReasoner*, of the cognitive processes that underlie these so-called ‘immediate’ inferences. The account is based on the assumption that when individuals understand discourse, they construct discrete mental simulations, i.e., mental models, of the assertions in the discourse. To draw conclusions, reasoners describe the relation between the individuals in the models and, if they are prudent, they search for alternative models to corroborate or refute a conclusion. We describe an experiment in which participants’ carried out a series of immediate inferences, and present a simulation that predicts the accuracy and latency of their responses.

Keywords: quantifiers, mental models, *mReasoner*, immediate inferences

Introduction

Reasoners can make immediate, rapid inferences from a single quantified assertion such as, *None of the Xs are Ys*. For instance, if they know that *none of the lawyers in the room are men*, they might refrain from asking any of the men in the room for legal advice, because they can infer:

1. None of the lawyers are men.
2. Therefore, none of the men are lawyers.

The inference is *valid* because its conclusion must be true given that its premise is true (Jeffrey, 1981, p. 1). It is also easy to make in comparison with more complex reasoning problems, such as syllogisms based on two quantified premises (for a review, see Khemlani & Johnson-Laird, in press a). Psychologists have investigated immediate inferences for many years (e.g., Begg & Harris, 1982; Newstead & Griggs, 1983; Wilkins, 1928), but have yet to resolve how logically untrained individuals make them. We have accordingly formulated a theory based on mental models and implemented it computationally in a unified model-based reasoning system called *mReasoner*.

In the present paper, we outline the theory and derive some novel predictions from it. We then report the results of a study that tested these predictions, and we show how the theory provides an satisfactory process model of individual performance.

Immediate inferences

In immediate inferences based on singly-quantified assertions, we studied 4 different *moods* for the premise:

All the Xs are Ys
Some of the Xs are Ys
None of the Xs are Ys
Some of the Xs are not Ys

and 8 different sorts of conclusion (4 moods by 2 *figures*, i.e., arrangements of terms ‘X’ and ‘Y’). Therefore, there are 32 possibly immediate inference problems based on these premises. The reasoner’s task was to assess a given conclusion’s status with respect to the premise, i.e., whether the conclusion *must* be true or whether it *might possibly* be true. Hence, it must be the case that some of the students are Virginians given than All the Virginians are students. And, it is possible but not necessary, that some of the students are not Virginians.

A robust theory of immediate inference should specify an algorithm that accounts for how individuals represent quantified assertions, how they assess whether a conclusion is at least possible, and how they decide whether it holds of necessity. It should also explain the relative difficulty of the various sorts of immediate inference, i.e., both the accuracy of participants’ conclusions and the latency of the correct conclusions. We developed such a theory as part of a general account of quantificational reasoning, and describe its assumptions below.

Mental models and quantifiers

Quantified assertions such as *None of the Xs are Ys* are can be treated as referring to relations among the set of Xs and the set of Ys (see, e.g., Cohen & Nagel, 1934, p. 124-5). Psychological theories of how quantifiers are interpreted follow suit (but cf. Braine & O’Brien, 1998; Rips, 1994), though they differ in the way they treat the relations between the sets. For instance, some theorists make use of diagrammatic representations to handle relations (Ceraso & Provitera, 1971; Erickson, 1974; Ford, 1995; Newell, 1981), others rely on formal rules of inference (Geurts, 2003; Guyote & Sternberg, 1981; Politzer, van der Henst, Luche, & Noveck, 2006; Stenning & Yule, 1997); and yet others analyze sets in terms of simulated possibilities, i.e., mental models (Bucciarelli & Johnson-Laird, 1999; Johnson-Laird & Byrne, 1991; Polk & Newell, 1995). The psychological systems can all account for how individuals make valid deductions, however few of them can account for the differences in relative difficulty between various reasoning problems. The present theory relies on mental models to explain the processes that give rise to valid and invalid

responses, as well as the differences in difficulty between various sorts of immediate inference.

The mental model theory proposes that when individuals comprehend discourse, they construct simulations of the possibilities consistent with the contents of the discourse (Johnson-Laird, 2006). The theory depends on three main principles: 1) Individuals use a representation of the meaning of a premise, an *intension*, and their knowledge, to construct mental models of the various possibilities to which the premises refer. 2) The structure of a model corresponds to the structure of what it represents (see Peirce, 1931-1958, Vol. 4), and so mental models are iconic insofar as possible. 3) The more models a reasoner has to keep in mind, the harder an inference is. On a model-based account, a conclusion is *necessary* if it holds in all the models of the premises and *possible* if it holds in at least one model of the premises.

Inferential processes

The theory proposes that the reasoning system can carry out three processes whenever individuals reason about whether a given conclusion follows from premises. First, the parser constructs a representation of the meaning of each premise (an *intensional* representation) based on a linguistic analysis. Second, the system uses the intension to construct a model of the situation to which the assertion refers (an *extensional* representation). Third, the system checks whether the given conclusion holds in the model. These three processes are carried out in “system 1” (see, e.g., Evans, 2003, 2007, 2008; Johnson-Laird, 1983, Ch. 6; Kahneman, 2011; Sloman, 1996; Stanovich, 1999; Verschueren, Schaeken, & d’Ydewalle, 2005) because they yield rapid responses from a single mental model.

The theory also postulates an advanced set of “system 2” processes, which are used to evaluate and, if necessary, to correct initial inferences. They search for alternative models of the premises in which both the premise and the conclusion are true. We explain how these processes work in the following section.

mReasoner: A computational theory of reasoning

mReasoner v0.9 is a new, unified computational implementation of the mental model theory of reasoning. It implements three general systems:

- a) A linguistic system for parsing premises and building up intensional representations. This system’s purpose is to map an assertion’s syntax to an underlying semantic form (the intension).
- b) An intuitive system (1) for building an initial extensional representation, and drawing rapid inferences from that representation.
- c) A deliberative system (2) for more powerful recursive processes that search for alternative models. This system can manipulate and update the representations created in System 1, and it can modify conclusions, but

it too can fall prey to systematic errors (Johnson-Laird & Savary, 1999; Khemlani & Johnson-Laird, 2009).

System 1 is faster than system 2, and as a result it is more prone to errors. Below, we describe the various processes that each system implements.

The linguistic system

Parsing statements into intensions. An *intensional* representation is composed from the meanings of words and the grammatical relations amongst them. The first process in mReasoner is a shift-and-reduce parser (Hopcroft & Ullman, 1979) that uses a context-free grammar and a lexicon to compose the intensional representations of a sentence. We make no claims about the psychological reality of a shift-and-reduce parser or a context-free grammar, which we adopt for convenience. The grammatical rules and lexical entries consist of a word (such as “all”), its part of speech (“determiner”), and a specification of its semantics. The parser applies the appropriate semantic rule, which matches the syntactic rule it has used, to construct an intension of the current constituent of the sentence. A key assumption is that the semantics of a quantified assertion sets the values of parameters that constrain the construction of models.

The present set of parameters is presented in Table 1 for the assertions that occur in the immediate inferences under investigation, and for two representative examples of other sorts of quantified assertions: *Neither A is a B*, and *Five A are B*. In sum, intensions are collections of parameters that, as a whole, specify the semantics of an assertion.

Table 1: A summary of mReasoner’s parameters in the intensions of different sorts of singly-quantified assertion. The parameters are as follows: i) the cardinality of the overall set of entities and its boundary conditions; ii) the cardinality of the set referred to by the quantifier (e.g., the As); iii) the boundary on the two sorts of cardinality as specified in (i) and (ii); iv) the polarity of the determiner; and v) the universality of the quantifier. Where relevant, ‘?’ signifies a default value that can be modified provided it meets the boundary conditions in (i) and (iii). A sixth parameter specifies the set-theoretic relation between the As and the Bs, and for all the examples below, the parameter is set to the inclusion relation, which specifies that the As are included within the set of Bs in the manner described by the other parameters.

Assertion	The five parameters in singly-quantified intensions				
	i	ii	iii	iv	v
All As are Bs	?4 ≥ 1	?4	ii = i	positive	universal
Some As are Bs	?4 ≥ 1	?2	0 < ii ≤ i	positive	particular
No As are Bs	?4 ≥ 1	?4	ii = i	negative	universal
Most As are Bs	?4 ≥ 1	?3	½ ii < ii < i	positive	particular
Neither A is a B	2	2	ii = i	positive	universal
Five As are Bs	5	5	ii = i	positive	universal

System 1

Model building. The system uses the intension of a premise to build an initial model, and it updates that initial model if additional premises occur. The model is built in accordance with the parameters of the intension. The system begins by building a model with a small set of individuals. For example, the model of *All the artists are bohemians* is built by first constructing a set of artists:

artist
artist
artist
artist

In the diagram above, each row represents an individual with the property of being an artist, and so the model as a whole represents a finite number of individuals. Mental models are representations of real individuals, not letters or words, which we use here for convenience. The inferential system in mReasoner is able to treat the model above as representing “all artists” and not, say, “exactly four artists” because it has access to the intension of the premise, which constrains the possible interpretations of models and therefore the possible modifications to models.

The intension of *all the artists are bohemians* also specifies the number of artists that are also bohemians. The model is updated accordingly:

artist	bohemian

At this point, the premise has been represented, and so the system assesses whether the given conclusion is true in the initial model.

Assessing initial conclusions. When reasoners have to assess a given conclusion, the system inspects the initial model to verify that the given conclusion holds or does not hold. For instance, suppose that reasoners are asked to decide whether it is possible that *some bohemians are not artists* given the previous premise. From the model above, the system initially responds in the negative, i.e., the putative conclusion is impossible. The process is simple, and the response is rapid. But, as we show in the next section, it is also fallible.

In many experiments and in daily life, reasoners have to draw their own conclusions. mReasoner accounts for this process too. The model above appears to support any of the following conclusions:

All the artists are bohemians
All bohemians are artists
Some artists are bohemians
Some bohemians are artists
Four artists are bohemians

But, the theory assumes that reasoners prefer to scan their initial model in systematic ways, and the computational system implements several heuristics that explain the general biases reasoners exhibit when they draw conclusions from inspecting a model of two quantified premises. Researchers often place heuristics at the forefront of theories of reasoning (e.g., Chater & Oaksford, 1999), but until now proponents of the model theory have downplayed their application. To unify model-based accounts of reasoning with heuristic-based systems, heuristics play a central role in inferring an initial conclusion (see Khemlani, Lostein, & Johnson-Laird, in press b, for an extended discussion).

The system’s ability to assess and generate initial conclusions is fallible. For instance, one can indeed show that *some of the bohemians are not artists* is possible, though the system infers initially that it is impossible. To revise its initial conclusion, the system needs to find an alternative model in which both the premise and conclusion hold. We turn to the final process in mReasoner to explain how such a model is found.

System 2

Searching for alternative models. In the preceding section, we focused on how mReasoner assesses conclusions based on an initial model. However, the conclusions it draws can be invalid. System 2 attempts to revise initial conclusions by searching for alternative models. To do so, it uses three separate operations: *adding* properties to individuals, *breaking* one individual into two separate individuals, and *moving* properties from individual to another (see Khemlani & Johnson-Laird, under review). These operations correspond to those that naïve participants spontaneously adopt when they reason about syllogisms (as evidenced by their manipulations of external models, see Bucciarelli & Johnson-Laird, 1999). In the earlier example, the system finds an alternative model by adding a new individual to the initial model to yield:

artist	bohemian
	bohemian

The new individual, who is bohemian but not an artist, and the resulting model refutes the conclusion, *All the bohemians are artists*. But, the conclusion, *Some of the bohemians are artists*, still holds, and no model refutes it.

Predictions. The theory and its computational implementation distinguish between the relative difficulty of three sorts of immediate inference:

- zero-model inferences
- one-model inferences
- multiple-model inferences

Zero-model inferences are those in which the conclusion is identical to the premise, and so individuals needn't even build a model to be able to solve the problem. For instance, consider the following problem:

All the aldermen are barters.
Is it possible that all the aldermen are barters?

The reasoner should realize that the answer is true immediately; however, individuals should nevertheless need to build intensions out of the assertions, and they need to establish a set of goals in order to infer a conclusion.

One-model inferences are those in which the conclusion holds in the initial model of the premise, and so individuals can rapidly determine that an assertion is possible. For example:

All the aldermen are barters.
Is it possible that some of the barters are aldermen?

Reasoners have to construct intensions, use them to build a model, and to evaluate the truth of the conclusion in the model.

Finally, multiple-model inferences are those in which the conclusion holds in an alternative model of the premise. For instance:

All of the aldermen are barters.
Is it possible that some of the barters are not aldermen?

The theory predicts that zero-model inferences should be easier than one-model inferences, and one-model inferences should be easier than multiple-model inferences. Likewise, the computational model predicts that individuals should respond faster to zero-model than one-model than multiple-model inferences. The predictions are unique to mReasoner and the model theory, because the theory proposes that one of the most important factors in judging the relative difficulty of different inferences is the number of models people have to construct. We ran an experiment to test these two rank-order predictions.

Experiment

A typical trial in the experiment was:

All the artists are bakers.
Is it possible that all of the bakers are artists?

The experiment examined all 32 possible sorts of problem, but we focused our analysis on only the 22 logically valid inferences. The inferences comprise 4 zero-model problems, 12 one-model problems, and 6 multiple-model problems.

Method

Participants. 26 participants completed the study for monetary compensation on Mechanical Turk, an online platform hosted by Amazon.com. None of the participants

reported having had any training in logic, and they were all native speakers of English.

Design and materials. The participants carried out all 32 problems (4 sorts of premise x 8 sorts of conclusion), and they responded "yes" or "no" to a conclusion about a possible conclusion to each problem. The contents of the problems were based on nouns referring to common vocations. We devised a list of 32 pairs of such vocations, which we assigned at random to the problems to make two separate lists. The problems were presented to each participant in a different random order.

Procedure. The study was administered using an interface written in PHP, Javascript, and HTML. On each trial, participants read the premise, and, when ready, they pressed a button marked "Next", which replaced the premise with a question concerning the immediate inference, e.g., "Is it possible that all the bakers are artists?" They responded by pressing one of two buttons labeled, "Yes, it's possible" and "No, it's impossible". The program recorded whether or not their response was correct, and its latency (in s). The instructions stated that the task was to respond to questions about a series of assertions concerning what was possible given the truth of the assertion. The participants carried out three practice trials in order to familiarize themselves with the task before they proceeded to the experiment proper.

Results and discussion

The results corroborated the theory's predictions of difficulty, and they yielded the following trend: reasoners were 98% correct for zero-model problems, 84% correct for one-model problems, and 70% correct for multiple-model problems (Page's trend test, $L = 340.0$, $z = 3.88$, $p < .0001$). mReasoner predicted the participants' accuracy well, $R^2 = .98$.

The mean latencies also corroborated the predicted trend: 4.30 s for zero-model problems, 5.17 s for one-model problems, and 5.41 s for multiple-model problems (Page's trend test, $L = 336.0$, $z = 3.33$, $p < .0005$). mReasoner's predictions of accuracy likewise explained a significant portion of the latency variance, $R^2 = .76$.

We found a good fit between mReasoner's predictions and the data from the 22 individual problems, where any significant correlation suggests a good fit. The number of models correlated with participants' accuracy, $R^2 = .36$. And the system's latency predictions adequately fit the latencies across the problems, $R^2 = .24$. The fit could be improved further, however, and we suggest several ways of proceeding in the General Discussion.

General Discussion

The computational theory, mReasoner, simulates the construction of mental models in order to draw immediate inferences from singly-quantified premises. The theory uniquely predicts that individuals should be faster and more accurate when an inference can be drawn from an identity in

intensions. They should be next fastest and accurate when an inference can be drawn from the initial model constructed in system 1. And they should be slowest and least accurate when an inference can be drawn only from the discovery of an alternative model constructed in system 2. The predictions are a result of assumption of the mental model theory of reasoning: the more models you need, the more difficult a problem becomes. These rank-order predictions were borne out in the data from an experiment that tested all 22 valid inferences about possible conclusions in the set of 32 inferences. The immediate inferences were easy: people answered correctly 83% of the time. Yet they also revealed subtle differences in the difficulty between the various sorts of problems along the lines predicted by the theory.

The system we describe is limited, however, and it can be improved to yield a more fine-grained processing account of the data. We suggest two separate ways of proceeding. One way to improve the fit of the system is to make the system sensitive to the direction in which it scans models. For instance, if reasoners read a particular premise, e.g., *all artists are bohemians*, they may be biased to scan the model in the opposite directions by considering bohemians before artists. This *figural* bias is widely documented in syllogistic reasoning (Khemlani & Johnson-Laird, in press a) and it is likely to make a difference when reasoning about immediate inferences as well.

mReasoner could also place differential costs on the underlying processes of each of its three systems. The linguistic system, system 1, and system 2 are groups of interrelated processes, and for simplicity, mReasoner treats each of the processes as though it should place the same temporal cost on the inference as a whole. The processes are likely to place different costs on the system, however, and future versions of the theory might investigate the low-level mechanisms that give rise to such costs (cf. Khemlani & Trafton, under review).

Immediate inferences have been restricted to the study of syllogistic assertions, e.g., those that make use of the determiners *all*, *some*, and *none*. However, one major advance of the system we describe is that it can be used to make predictions if a broader range of inferences. Consider the following inference:

Most politicians are wealthy.
Is it possible that most wealthy people are not politicians?

The answer, like many of the inferences above, is easy: given the first premise, it is indeed possible that most wealthy people are not politicians. However, the inference likely engaged a search for alternative models, and so it should take reasoners longer to make it than a problem in which the conclusion follows straight from the premise:

Most politicians are wealthy.
Is it possible that some wealthy people are politicians?

Indeed, the former inference might have required a little thought, whereas the latter one may have felt “obvious.” A viable theory of immediate inferences should be able to account for any difficulty between the two problems, and at present, mReasoner is the only system that can do so. In the same vein, the system can be used to make predictions about problems that make use of statements such as:

Neither of the Xs is a Y
At most five of the Xs are Ys
More than a third of the Xs are Ys
Five of the Xs are not Ys

and so it is more general than theories restricted to Aristotelian syllogisms.

In sum, mReasoner is a new, unified computational cognitive model of deductive reasoning. It is based on the mental model theory of human reasoning, and so its primary prediction is that problems that require multiple models are difficult and take longer than problems that require only one model.

Acknowledgments

This research was funded by a National Research Council Research Associateship awarded to SK; ONR Grant #s N0001412WX30002 and N0001411WX20516 to JGT; and NSF Grant No. SES 0844851 awarded to P.J.L. We are also grateful to Len Breslow, Magda Bugajska, Hua Gao, Tony Harrison, Cathy Haught, Laura Hiatt, Gorka Navarrete, Marco Ragni, and Tobias Sonntag for their helpful comments. The views and conclusions contained in this document do not represent the official policies of the US Navy.

References

- Begg, I., & Harris, G. (1982). On the interpretation of syllogisms. *Journal of Verbal Learning and Verbal Behavior*, 21, 595–620.
- Braine, M.D.S., & O'Brien, D.P., Eds. (1998). *Mental logic*. Mahwah, NJ: Erlbaum.
- Bucciarelli, M., & Johnson-Laird, P. N. (1999). Strategies in syllogistic reasoning. *Cognitive Science*, 23, 247–303.
- Ceraso, J., & Provitera, A. (1971). Sources of error in syllogistic reasoning. *Cognitive Psychology*, 2, 400–410.
- Cohen, M.R., & Nagel, E. (1934). *An introduction to logic and scientific method*. London: Routledge & Kegan Paul.
- Erickson, J.R. (1974). A set analysis theory of behavior in formal syllogistic reasoning tasks. In R. Solso (Ed.), *Loyola symposium on cognition*, Vol. 2 (pp. 305–330). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Evans, J.St.B.T. (2003). In two minds: Dual process accounts of reasoning. *Trends in Cognitive Sciences*, 7, 454–459.
- Evans, J. St. B. T. (2007). *Hypothetical thinking: Dual processes in reasoning and judgement*. Hove, UK: Psychology Press.

- Evans, J. St. B. T. (2008). Dual-processing accounts of reasoning, judgment and social cognition. *Annual Review of Psychology*, *59*, 255-278.
- Ford, M. (1995). Two modes of mental representation and problem solution in syllogistic reasoning. *Cognition*, *54*, 1-71.
- Geurts, B. (2003). Reasoning with quantifiers. *Cognition*, *86*, 223-251.
- Guyote, M.J., & Sternberg, R.J. (1981). A transitive-chain theory of syllogistic reasoning. *Cognitive Psychology*, *13*, 461-525.
- Hopcroft, J.E., & Ullman, J.D. (1979). *Formal Languages and Their Relation to Automata*. Reading, MA: Addison-Wesley.
- Jeffrey, R. (1981). *Formal logic: Its scope and limits* (2nd Ed). New York: McGraw-Hill.
- Johnson-Laird, P.N. (1983). *Mental Models*. Cambridge: Cambridge University Press. Cambridge, MA: Harvard University Press.
- Johnson-Laird, P. N. (2006). *How we reason*. Oxford, UK: Oxford University Press.
- Johnson-Laird, P.N., & Byrne, R.M.J. (1991). *Deduction*. Hillsdale, NJ: Erlbaum.
- Kahneman, D. (2011). *Thinking, fast and slow*. New York, NY: Farrar, Strauss, Giroux.
- Khemlani, S., & Johnson-Laird, P.N. (2009). Disjunctive illusory inferences and how to eliminate them. *Memory & Cognition*, *37*, 615-623.
- Khemlani, S., & Johnson-Laird, P.N. (in press a). Theories of the syllogism: A meta-analysis. *Psychological Bulletin*.
- Khemlani, S., & Johnson-Laird, P.N. (in press b). The processes of inference. Manuscript under submission. *Argument and Computation*.
- Khemlani, S., Lotstein, M., & Johnson-Laird (under review). A unified theory of syllogistic reasoning. Manuscript under submission.
- Khemlani, S., & Trafton, J.G. (under review). mReactr: A computational theory of deductive reasoning. Manuscript under review.
- Newell, A. (1981). Reasoning, problem solving and decision processes: the problem space as a fundamental category. In R. Nickerson (Ed.), *Attention and Performance*, Vol. 8 (pp. 693-718). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Newstead, S.E., & Griggs, R.A. (1983). Drawing inferences from quantified statements: A study of the square of opposition. *Journal of Verbal Learning and Verbal Behavior*, *22*, 535-546.
- Peirce, C.S. (1931-1958). *Collected papers of Charles Sanders Peirce*. 8 vols. Hartshorne, C., Weiss, P., & Burks, A. (Eds.) Cambridge, MA: Harvard University Press.
- Politzer, G., van der Henst, J.B., Luche, C.D., & Noveck, I.A. (2006). The interpretation of classically quantified sentences: A set-theoretic approach. *Cognitive Science*, *30*, 691-723.
- Polk, T.A., & Newell, A. (1995). Deduction as verbal reasoning. *Psychological Review*, *102*, 533-566.
- Slooman, S.A. (1996). The empirical case for two systems of reasoning. *Psychological Bulletin*, *119*, 3-22.
- Stanovich, K.E. (1999). *Who is rational? Studies of individual differences in reasoning*. Mahwah, NJ: Erlbaum.
- Stenning, K., & Yule, P. (1997). Image and language in human reasoning: A syllogistic illustration. *Cognitive Psychology*, *34*, 109-159.
- Verschuere, N., Schaeken, W., & d'Ydewalle, G. (2005). A dual-process specification of causal conditional reasoning. *Thinking & Reasoning*, *11*, 278-293.
- Wilkins, M. C. (1928). The effect of changed material on the ability to do formal syllogistic reasoning. *Archives of Psychology*, *16*, No. 102.

Cognitive Robotics: Analysis of Preconditions and Implementation of a Cognitive Robotic System for Navigation Tasks

Stefano Bennati [benna86@gmail.com]

Università degli Studi di Brescia
Brescia, Italy

Marco Ragni [ragni@cognition.uni-freiburg.de]

Department of Cognitive Science, Friedrichstrasse 50
Freiburg 79098, Germany

Abstract

Cognitive robotics is a fascinating field in its own right and comprises both key features of autonomicity and cognitive skills like learning behavior. Cognitive architectures aim at mirroring human memory and assumptions about mental processes. A robot does not only extend the cognitive architecture regarding real-world interaction but brings new and important challenges regarding perception and action processes. Cognitive robotics is a step towards embodied cognition and may have influences not only on cognitive science but on robotics as well. This paper presents an integration of the cognitive architecture ACT-R on a simple but programmable robotic system. This system is evaluated on a navigation experiment.

Keywords: Cognitive Science; Robotics; Mindstorms; ACT-R; Navigation

Introduction

From the early beginning of robotics one line of research has tried to bring human cognition and robotics closer together Brooks et al., 1999. Nowadays, technological progress in the field of robotics and the development of cognitive architectures allows for a leap forward: A robot able to navigate an environment, with the ability to learn and a human-like attention shift.

This new and exciting field is sometimes referred to as *Cognitive Robotics*¹. This combination of two fields leads to a number of important research questions: What are the immediate advantages of cognitive robotics (a term we will use in the following for a robot controlled by a cognitive architecture) over classical robotics? Is the cognitive architecture (which partially is able to simulate human learning processes) restricting or improving navigation skills? In cognitive science new research focuses on *embodied cognition*.

Embodied cognition claims that understanding (especially of spatial problems) is derived from the environment Anderson, 2003. In other words, cognition is not independent on its physical realization. The study described in Buechner et al., 2009 used a virtual reality environment. Participants had to navigate through a labyrinth in the ego-perspective and had to find an initially specified goal (red dot). The study identified recurrent navigation strategies (which we introduce later) used by the subjects.

¹"Towards a Cognitive Robotics", Clark, Andy <https://www.era.lib.ed.ac.uk/handle/1842/1297>

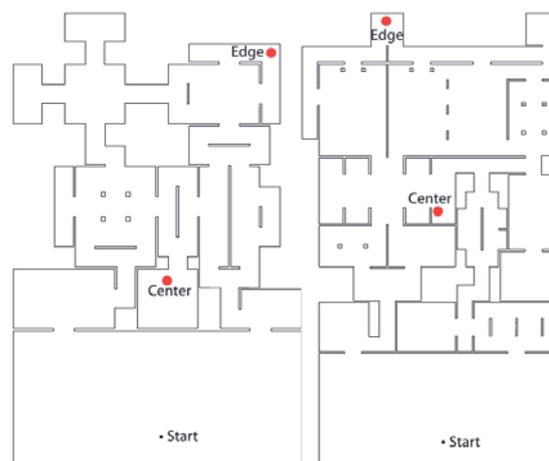


Figure 1: Layouts of two mazes: The task is to find the target object (red dot) at the edge or center Buechner et al., 2009.

Modeling navigation tasks, for instance in labyrinths, still poses a challenge for cognitive architectures: Although they can model decision processes they typically abstract from metrical details of the world, from sensor-input, and from the integration processes of environmental input to actions (like move operations). Robotics, on the other hand, has typically captured all of these aspects, but does not necessarily make use of human-like learning and reasoning processes or even try to explain human errors or strategies.

Compared to humans a robotic agent has limited perceptions and capabilities. On the other hand, it can teach us something about the relevant perceptions that are already sufficient for the robotic agent to perform successfully. For instance, in the navigation task above (e.g., cp. Fig. 1), the distance from the wall (in the direction it is facing) and the color of the floor the robot is standing on are sufficient.

Much research is being done in the field of the cognitive robotics; some prototypes of cognitive robots have already been built². This research concentrates on the human-robot and robot-environment interaction, allowing the robots to rec-

²e.g., Cognitive Robotics with the architecture ACT-R/E <http://www.nrl.navy.mil/aic/iss/aas/CognitiveRobots.php>

ognize and interact with objects and people through their visual and auditory modules Sofge et al., 2004. The architecture proposed contains *Path Planning and Navigation routines* based on the Vector Field Histogram Borenstein and Koren, 1991 that allow the robot to navigate avoiding obstacles and explore the environment. Unlike the architecture proposed, the objective of this paper is to implement a *Cognitive Navigation System* which is completely based on ACT-R and takes advantage of its cognitive features like *Utility Learning*, that allows Reinforcement Learning Russell and Norvig, 2003, p.771 on productions. No other softwares than ACT-R will be used to control the robot. That experiment has the merit of have taken the first steps towards interfacing ACT-R with a mobile robot, but the data is still incomplete and tries to combine as many different abilities as possible (from natural language processing, to parallel computing) in a manner that is likely more complex than necessary. Our approach starts at the other end, taking only those aspects and sensor data into account that are necessary to perform the task – the most simple robot. Other research studied the possibility of interfacing ACT-R with a robot and giving it direct control over the robot’s actions. That effort produced an extension of ACT-R called *ACT-R/E(mbodyed)* Trafton, 2009; Harrison and Trafton, 2010. ACT-R/E contains some new modules that act as an interface between the cognitive model and the *Mobile-Dexterous-Social (MDS)* robot Breazeal et al., 2008, allowing it to perceive the physical world through a video camera. However, no navigation was investigated, as the robot did not navigate an environment. In both this and our implementation ACT-R has been extended. The vast difference between the two is the smaller amount of changes made to the standard ACT-R by our implementation, due to the sensors’ higher complexity in the MDS.

Consequently this article investigates, first, how to control a robot through ACT-R and, second, if this *cognitive robot* shows human-like behavior while navigating and searching for a goal, e.g., an exit from the maze. The paper is structured in three parts: The first part – The Elements – contains a brief description of ACT-R, the robot and its features. The second part – the Integration – describes the software that connects ACT-R with the robot, through which perceptions can reach the cognitive architecture and actions the actuators. The third part – the Evaluation – tests the robot on a navigation task and analyzes the results.

The Elements

The Robotic System

Our device of choice is a standard Mindstorms³ class robot (cf. Fig. 2). It consists of a central “brick” that contains the processor and the batteries. At this core component several peripherals can be attached. It supports up to three step-to-step engines and up to four sensors.

³This type of robot is produced by The Lego Group



Figure 2: The Mindstorms class robot. It is equipped with a color, ultrasonic and two touch sensors.

Chassis design. The chassis is the structure to which the central brick, the motors and the sensors are attached. To limit odometry errors two fixed, independent driving wheels have been used, instead of caterpillar tracks. A third central fixed idle wheel allows the robot to keep its balance. The structure resembles a differential drive robot, with the exception that the third wheel cannot pivot. Removing the tire from the wheel allows the robot to turn without too many difficulties.

Sensors. A robot can be equipped with several kinds of sensors: from a simple sound or light sensor, to more complex ones like a compass or webcam. Our design makes use only of the most basic sensors: a *touch sensor* activated by pressure, a *color sensor* capable of recognizing light intensity or six tonalities of color, and an *ultrasonic sensor* for distance measurements. The color sensor and the ultrasonic sensor are fixed to the front, while a touch sensor is placed on each side. Both touch sensors are linked to a structure that covers the front of the robot on that side. They are not used during navigation but to stop the robot when it touches a wall.

Integrating Architecture and Robot

The first step towards a *cognitive robot* is to create a working interface between the ACT-R framework and the NXT platform. This interface allows an ACT-R model to control the robot and to receive sensor inputs from the robot. Due to the modular structure of ACT-R, this interface is composed of modules.

Low level lisp function. On the basis of these modules there is a library called *nxt-lisp* Hiraishi, 2007 that provides low-level lisp functions to execute simple tasks like interrogate a sensor or move a motor. The capabilities of this library are the following: it can read information about the environment from the light sensor, the distance sensor and the touch sensor; it can make the robot perform some actions, like move its motors, turn on and off the light and play some sounds; it can also check the robot’s internal state, querying the battery or the motors about their states. The library did not have the support for the color sensor, so it has been extended to support that sensor, necessary for our purposes, as well.

Extending ACT-R

The interface is composed by several modules with different functions, this step has been taken to keep things simple and to follow the general design of ACT-R. The lisp library has been bundled in the code and its functions are called by these modules to control the robot.

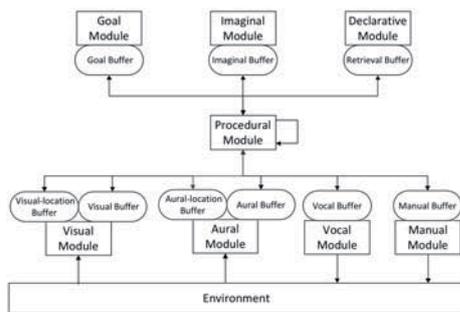


Figure 3: The ACT-R modules.

Motor module. This module controls the engines, when a request is made to its buffer the module responds, activating or deactivating the specified motors.

Touch module. This module controls the touch sensors, it can control up to four sensors but the default value is two. This module has a buffer called `nxt-touched`, but it is only used to query the module. If the `nxt-touched` buffer receives a query on the touched slot, with value `true`, the query returns `true` if the sensor reports a pressure.

Vision module. This module controls the color sensor. It has a buffer called `nxt-visual`, but it does not have any other purpose than querying the module. If the `nxt-vision` buffer receives a query on the touched light, with value `on`, it will turn the sensor on and schedule a polling of the sensor every 50ms. If the value is `off`, the sensor will be turned off and the polling stopped. When the sensor identifies a change in color, or the color blue, it draws on the visicon, the standard ACT-R visual input, a letter of the same color. This action triggers a standard ACT-R procedure that will allow the model to realize that a change has happened.

Distance module. This module controls the ultrasonic sensor. It has a buffer called `nxt-distance`. When a request is made to this buffer, with a chunk of type `obstacle`, the module reads the distance from the sensor and updates the chunk in its buffer with the read value. It uses two productions to obtain the value, the first production fires a request and the second reads the results from the `nxt-distance` buffer.

Evaluation: Navigation in a Labyrinth

We decided to test the cognitive robot on several self-built labyrinths. The environment is perceived by the robot through its sensors, the robot can perceive the presence of an obstacle, the color of the ground that is used to discriminate between obstacles (a wall, a junction, the goal, or a clear way), and the distance to the next wall. It is important to keep in mind that the robot has no a priori knowledge of the environment it is operating in. It will explore the labyrinth like humans would – this is a classical example of contingency problem Russell and Norvig, 2003, p.80. The only two types of information that the robot can obtain from its sensors are the presence/absence of an obstacle close by, and its distance from the next wall. With this information the robot must avoid walls and be able to choose in which direction to turn when it finds a junction. A human being can approximately measure this distance in two ways: looking at the walls a person can guess its distance from it, like the robot does with its ultrasonic sensors, and through his sense of touch, the same as the touch sensors equipped on the robot.

The Model

The model implements the *Active Reinforcement Learning* algorithm Russell and Norvig, 2003, p.771, that uses the feedback on its performance to learn what actions are to be preferred. In case the available information is not enough to decide what is the best way, the model follows the perimeter strategy to navigate. An aleatory component can favour a known way over the entrance in a not yet explored branch. Thanks to its internal representation of the environment, the model is able to recognize the junctions it has visited before and remember their performance.

The internal representation

While exploring, the cognitive robot develops an internal representation of the environment in declarative memory. This representation contains all of the information that it knows about the already explored environment. For every step or turn the robot takes, information is stored in declarative memory in a chunk of type “movement,” those chunks have a slot called “direction” that encodes the direction of the very movement: This slot has the value “forwards”, unless the robot turns itself. In that case the slot will contain the name of the new orientation. Every movement chunk has another slot called “counter”, where a progressive number is stored. The numbers detail the sequence of movements of the current run. Through this trace, it is always possible to retrieve the direction it is currently facing. This is the only purpose of using these chunks during navigation. The maze is represented in declarative memory as a graph, whose nodes are the junctions. The model does not care about how long and how twisted the road from JunctionA to JunctionB is, that is unimportant during the decision process because it does not involve decision making. The only relevant information is the expected performance in taking a specific way, which is represented in declarative memory by a chunk of type “junction.”

This chunk identifies a specific junction by the values in four of its slots (north, east, south and west) and a direction by its “turn” slot. For every turn the model must know its goodness, this value is encoded in the “performance” slot. A smaller number implies a better performance, the number “-1” means that this direction was taken but not yet rated. Recognizing a junction allows the model to remember which routes it has already explored and the current best. With such a complete knowledge the model is able to find a shortest way to the goal. These two chunk types form the model’s *Knowledge Base*, through which the previous explorations can be reconstructed. The “movement” chunks contain the path history: They save, for every step, the direction in which the model was going; while the “junction” chunks compose the decisional history and save the order in which the junctions were taken and the performance value’s updates of every junction.

Algorithm

The robot uses a local search algorithm (cf. Fig. 4). The robot moves forward until a chunk is present in the visual-location buffer, then it reads the color and discriminates if it is an obstacle, a junction, the goal or a false alarm. In case of a false alarm it keeps going forwards. If a wall is detected there could be a curve right or left, or a dead-end. Special productions are called for a quicker response and the robot will follow the curve or, in case of a dead-end, go back. If a dead-end is encountered, the last junction is marked with a low performance value, so that in future the model will avoid that direction. When a junction is detected, with or without a wall on the front, the robot turns itself in the four directions and measures the distance with its ultrasonic sensor. Once the distances are retrieved, it tries to recognize this junction among the ones it has seen before. Many retrieval requests to the declarative memory are issued, by calling the same production many times, until the retrieval process fails. For every direction that is not retrieved, the model checks the corresponding distance. If it is less than a certain threshold it detects a wall and marks that direction as not selectable. Now, if there is still at least one direction that has not been tried yet, the model has two possibilities: The first possibility is to select that direction and explore a new branch of the labyrinth; if there is more than one, the model follows the perimeter strategy. In this way the maze, in the first stage of the exploration, is covered following the perimeter strategy. The second possibility is to select the way with the best performance among the directions that matched in declarative memory and go on a safe path. The decision between an enterprising and a conservative approach is taken randomly, with a probability of taking a conservative path proportional to the performance rating: The better the performance, the more likely the conservative approach is to be selected. This solution incentivizes the exploration in the first stage, when it is more likely that a shorter way can be found; and is more conservative in the end, when exploration will bring a minimal increment of performance. If all the selectable directions matched in declarative memory, the model chooses the

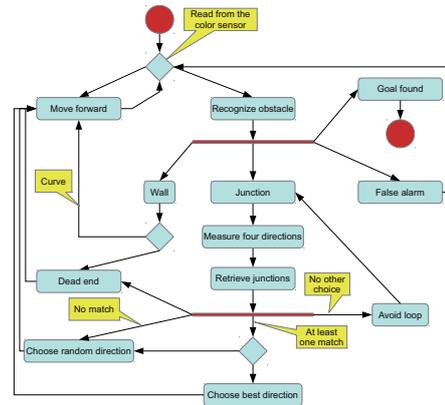


Figure 4: The local search algorithm implemented in ACT-R.

way with the best performance. If the junction does not have any selectable direction, because there are only walls or dead-ends, the branch is purged: the last junction, that brought the model to this dead-end, is selected and marked with a very poor performance value. In the particular case in which the only selectable direction has already been taken during the current run, the robot retraces its steps, a loop is detected. The last unrated junction that the robot took is marked as a dead-end, so that the next time it will not turn in that direction again, interrupting the loop. If a goal is detected all the junction chunks used in the last run are rated by a rating function that implements the *Policy-Iteration* routine Russell and Norvig, 2003, p.624. For every junction the rating function calculates the new performance value as the time difference between the goal discovery and the last use of the junction’s representation in declarative memory. If the old performance value is higher, or that junction had not been rated yet, the function updates the performance with the current value. After that the model starts again from the beginning, but this time it uses the accumulated experience during the previous explorations leading to better choices.

Distance computation

The Act-Rientierung Project Dietrich et al., 2011 implemented an algorithm that reconstructs the distance between non-adjacent waypoints, using the information about adjacent waypoints gleaned from exploration. It has been integrated in our model. We used junctions as waypoint because the environment and the perceptions did not leave other choice. The computation is made using chunks that represent numbers, adjacent numbers have high similarity and that leads to counting errors (reproducing human behavior). The model, during exploration, stores distance in steps of adjacent junctions in specific “waypoint” chunks. When the model finds the goal, it goes through the list of known junctions and calculates the distance to the goal for every pair of junctions. With consecutive runs the distance values may change, because of the retrieval errors in the counting process.

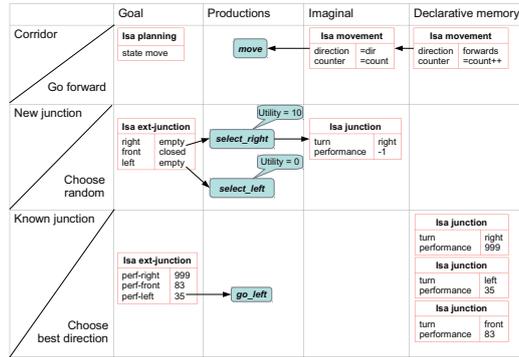


Figure 5: **Left:** An example of a test labyrinth. **Right:** Decision process for corridor, new and known junctions.

Evaluation

Several tests have been executed to evaluate the differences in performance that this new implementation brings. Those tests have been run in a simulated environment that reproduces exactly a real robot going through a real maze, with the advantages of being faster and not having to deal with odometry or measurement errors. To be able to compare the results the test has been set up as a replica of the Büchner experiment Buechner et al., 2009, recreating the same labyrinths inside the simulator (cf. Fig. 1); in both the configuration with the goal on the edge and in the center. The performance was measured in PAO (Percentage Above Optimum).

$$PAO = ((d_{walked} - d_{shortest}) / d_{shortest}) \times 100$$

The same unit of measurement was used in this experiment as well, so that the results of both can be easily compared. During every simulation, for each of the four environments, the model had a time limit of 4000 units of ACT-R's virtual time. Within this time constraint the model could find a stable suboptimal solution was found 84% of the time. The diagram in Fig. 6 shows the mean quality of the consecutive solutions.

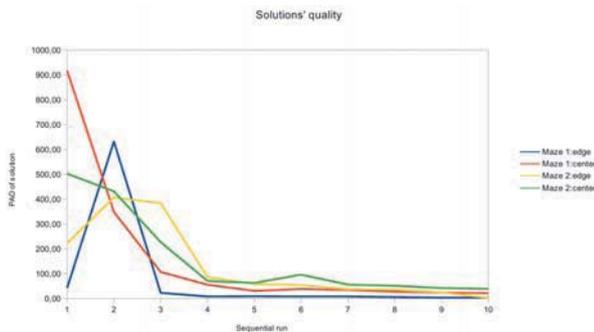


Figure 6: Behaviour of the model, showing a learning curve similar to the humans' Buechner et al., 2009.

The curves in Fig. 6 show similar learning behavior for all four labyrinths: The model starts with an uninformed exploration, according to the actual strategy, until it finds the goal. Then it starts an exploration phase that will complete when all possibilities have been tried and rated. During this phase

the model tries unexplored ways, needing in average more time to find the goal than the human counterpart. After less than 4 runs, on average, the model has gathered enough information about the environment and stabilizes on a suboptimal solution that is covered until the time elapses. In the experiment done by Büchner the participants shown a preference for the perimeter strategy to navigate in the maze and search for the goal. In the current experiment different strategies were tested:

Deterministic perimeter strategy. The first strategy uses a right-preference perimeter strategy. Each application of turn-right production rules yields a utility bonus, a smaller bonus is given for go-straight production while for turn-left no bonus is assigned. The test showed that the first run is completely deterministic and in every run the agent always take the same path, that implies a much smaller degree of choice during the rest of the exploration. Even if some differences can be seen during the exploration phase, every run leads to the same suboptimal solution. In conclusion the tests demonstrated that this strategy is too deterministic and dependent on the specific maze to be suitable for a real exploration task.

Random walk. Another strategy was a random strategy with utility learning, in the beginning all the productions have the same utility, which can change during the run according to the utility rewards gathered during the exploration. A positive utility reward is given by the action of finding the goal, with a negative reward for each dead-end. As expected from a random walk strategy, the performance is nearly identical for all four environments.

Gaussian perimeter strategy. The last strategy uses a random gaussian utility bonus, instead of fixed one. The more desirable is the action to perform, the higher the center of the gaussian will be. This bonus is added to the standard production's utility by the mean of the *utility-offsets* function Bothell, 2004, p.187. The gaussian functions are calculated by the *act-r-noise* function Bothell, 2004, p.138 with parameter $s = 0.5$, that correspond to a variance $\sigma^2 = 0,82246$.

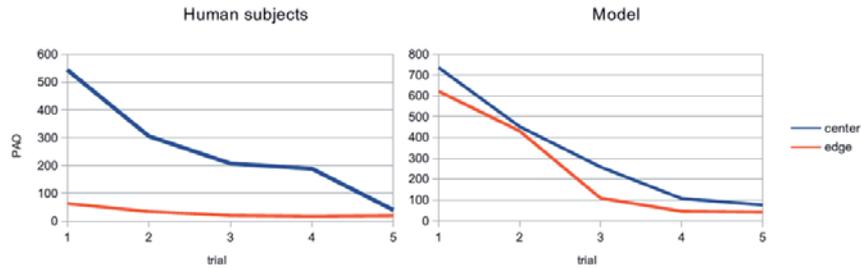


Figure 7: Human (left) and model performance (right) with random walk, the curves have a correlation of 0,962 and 0,954.

The three curves, identifying the action of turning left, going straight on and turning right, are centered respectively in 0, 2 and 4, the higher the center the more likely to happen. This method models well the human concept of perimeter strategy, not mere deterministic but a good compromise between following the rules and taking own initiative. The tests report that the perimeter strategy is chosen 47% of the time in the first maze and the 43% in the second, and that initial value and solution quality are close to the human's. Some differences are present in the execution, due to the model's intensive exploration, unlike the human subjects who prefer to travel along a safe path.

Conclusions & Future work

This experiment demonstrated that ACT-R can be successfully used in conjunction with a Mindstorms robot. We applied a bottom-up method starting with a restricted number of sensors and computational power, relying more on the cognitive aspects of ACT-R. Our cognitive model, implemented in ACT-R, successfully simulates the human behavior and learning strategies while navigating in a labyrinth, and shows human performances. The cognitive model could be further enhanced by implementing a better simulation of human memory, for example adding a utility threshold under which a chunk is not retrieved. If the model is able to forget the less used junctions it would behave more like a human being. Some tests in this direction have been executed, but to get valuable results more work is needed on the parameter's tuning. At the moment the model has a perfect memory and the only source of error is the ambiguity between junctions. Another interesting enhancement is to allow the model to randomly forget or switch junction performances, as well as the user direction, like humans tend to do when the quantity of information they have to remember is too great. This feature could be implemented using the similarity function embedded in ACT-R.

Acknowledgments

The authors are grateful to Dan Bothell for extensive discussions during the coding phase, Tasuku Hiraishi who developed the `nxt-lisp` library, and Simon Büchner for discussions about his study and human navigation in general. This work has been partially supported by the SFB/TR 8 "Spatial Cognition" within project R8-[CSPACE] funded by the DFG.

References

- Anderson, M. (2003). Embodied cognition: A field guide. *Artificial intelligence*, 149(1):91–130.
- Borenstein, J. and Koren, Y. (1991). The vector field histogram and fast obstacle-avoidance for mobile robots. *IEEE Journal of Robotics and Automation*, 7:278–288.
- Bothell, D. (2004). ACT-R 6.0 reference manual. Technical report.
- Breazeal, C., Siegel, M., Berlin, M., Gray, J., Grupen, R. A., Deegan, P., Weber, J., Narendran, K., and McBean, J. (2008). Mobile, dexterous, social robots for mobile manipulation and human-robot interaction. In *SIGGRAPH New Tech Demos*, page 27. ACM.
- Brooks, R., Breazeal, C., Marjanović, M., Scassellati, B., and Williamson, M. (1999). The cog project: Building a humanoid robot. In *Computation for metaphors, analogy, and agents*, pages 52–87. Springer-Verlag.
- Buechner, S., Hölscher, C., and Wiener, J. (2009). Search strategies and their success in a virtual maze. In *Proceedings of the 31st Annual Conference of the Cognitive Science Society*, pages 1066–1071.
- Dietrich, F., Kan, J., et al. (2011). ACT-Rientierung. <http://sourceforge.net/projects/act-rientierung/>.
- Harrison, A. and Trafton, J. (2010). Cognition for action: an architectural account for "grounded interaction". In *Proceedings of the 32nd Annual Conference of the Cognitive Science Society*, pages 246–251.
- Hiraishi, T. (2007). Nxt controller in lisp. <http://super.para.media.kyoto-u.ac.jp/~tasuku/index-e.html>.
- Russell, S. J. and Norvig, P. (2003). *Artificial Intelligence: a modern approach*. Prentice Hall, 2nd international edition edition.
- Sofge, D., Trafton, J. G., et al. (2004). Human-robot collaboration and cognition with an autonomous mobile robot. *Intelligent autonomous systems* 8, page 80.
- Trafton, J. (2009). An embodied model of infant gaze-following. Technical report, DTIC Document.

Accumulation of Evidence and Information Search in Experiential Decisions

Katja Mehlhorn (katjam@cmu.edu)

Varun Dutt (varundutt@cmu.edu)

Noam Ben-Asher (noamba@andrew.cmu.edu)

Cleotilde Gonzalez (coty@cmu.edu)

Dynamic Decision Making Laboratory, Carnegie Mellon University
4609 Winthrop Street, Pittsburgh, PA 15213 USA

Keywords: information search, decisions from experience, sampling paradigm, cognitive model, evidence accumulation.

Imagine you are about to write a paper for an upcoming conference. Before you decide how to write it, you are likely to read a number of related articles. How many articles do you read? When do you stop searching for more articles and start writing? While we know much about how people make decisions, little is known about the search process that precedes a consequential decision. Here, we analyze participants search behavior in a binary choice task and present a cognitive model that is able to explain both, the process of information search, as well as the subsequent consequential choice.

Information search in decisions from experience

In the classical decision-making literature, decision-making has often been studied by explicitly presenting the decision maker with the information relevant for the decision. Thus, this literature largely ignores the process of information search. Recently, experimental paradigms have studied experiential decisions and allowed the investigation of the search process before decisions are made. One prominent example is the sampling paradigm (e.g., Hertwig, Barron, Weber, & Erev, 2004). In this paradigm, participants are presented with two options (visualized as two buttons on the screen) that are associated with monetary payoff distributions. For example, in the decision problem shown in Figure 1, the left button (risky button) yields a high payoff (17.1) with 10% probability and a low payoff (6.9) with 90% probability. The right button (safe button) yields a medium payoff (8) with 100% probability. Participants are not explicitly told about the payoffs or their probabilities. Instead, they are asked to sample from both options until they feel confident to make a consequential choice for the option they prefer.

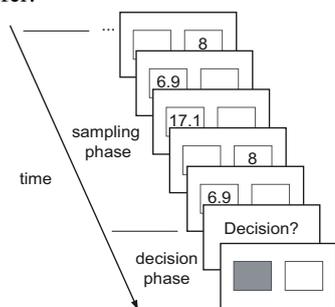


Figure 1. Procedure in the sampling paradigm.

Research has shown that, overall, people tend to take only a small number of samples (medians range from 9-19 in most studies; Erev et al., 2010; Hau, Pleskac, & Hertwig, 2010). This small sample size is surprising, given that the chance to correctly estimate the payoff distribution is likely to increase with the sample size (Gonzalez & Dutt, 2012). While the sample size seems to be affected by several characteristics of the decision maker, not much is known about how it is influenced by different properties of the task.

To learn more about when people stop searching for information, we analyzed data from the Technion Prediction Tournament (Erev, et al., 2010), which is the largest data set on the sampling paradigm currently available (79 participants, each solving 30 out of 120 problems). In agreement with the small sample size effect, Figure 2 shows that in the TPT data set the distribution of the sample size is heavily right-tailed. To take a closer look at how the sample size is affected by properties of the decision problems, we investigated two factors: experienced variability and payoff domain. As shown in Figure 3, the sample size increases when variability was experienced in a problem (i.e., when the risky button displayed two possible outcomes, as in Figure 1), compared to problems where no variability was experienced (the risky button displayed only one outcome); average medians: 11.1 vs. 15.3. Furthermore, the number of samples is lower if the observed payoffs constituted gains (as in Figure 1), rather than losses (where outcomes on both buttons were negative); average medians: 12.2 vs. 14.0. A linear mixed effects model, showed the effects of both factors to be significant (variability: $\beta=4.65$, $p<.001$; domain: $\beta=1.64$, $p=.001$).

Modeling information search

Recently, Gonzalez and Dutt (2011) have shown that a computational model based on instance-based learning theory (IBLT; Gonzalez, Lerch, & Lebiere, 2003) can not only explain how people make experiential decisions, but also explain how information is sampled before a consequential decision is made. The basic idea in the IBL model is that, during sampling, instances of the observed payoffs in both options are stored in memory. Behavior during sampling and at final choice depends on the experienced utility (or bended value) of the options. The experienced utility of an option is a function of its associated payoffs and the probability of retrieving these payoffs (i.e., instances) from memory, using a simplified ACT-R activation mechanism (Anderson & Lebiere, 1998).

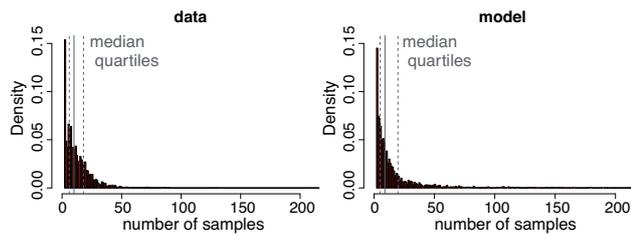


Figure 2. Distribution of the sample size. The x-axes are truncated at the maximum number of samples found in the human data (resulting in cutting off 2% of the tail produced by the model).

The IBL model explains, for example, how participants alternate between the choice options during sampling (for a detailed explanation of the model see Gonzalez & Dutt, 2011).

Although the IBL model is able to account for choices during sampling, the stopping point (i.e., the number of samples drawn during the sampling phase) was previously determined by a random draw from a geometric distribution function that was fitted to behavioral data in TPT. Here, we introduce a stopping mechanism in the model that is grounded in psychological literature related directly to information foraging. This mechanism is motivated from evidence accumulation models (for an overview see e.g., Ratcliff & Smith, 2004). It assumes that people accumulate experienced utilities until a decision criterion is reached. Once this criterion is reached, sampling is stopped and a decision is made based on the experienced utilities as in the original formulation of the IBL model. The revised model is thus identical to the model reported in Gonzalez and Dutt (2011), with the exception that the stopping point is now determined by an evidence accumulation process, rather than by a mathematical distribution function.

To fit the revised model, we calibrated the decision criterion (which consists of an upper bound for positive and a lower bound for negative experienced utilities), to the median number of samples from the human data. All other parameters were left at the original values reported in Gonzalez and Dutt (2011). More specifically, we calibrated the bounds by using a Genetic Algorithm (Holland, 1975) to fit the model's median sample size to the median sample size for half of the human data. The resulting bounds were randomly drawn from $U(0.0001, 14.18)$ for positive, and from $U(-24.18, -0.0001)$ for negative values. Then, we generalized the model to the other half of the data.

With the model merely calibrated to fit the median sample size, we then investigated whether it would reproduce the distribution of the number of samples, as well as the effects of the two task-relevant factors (identified above). As shown in Figure 2, the model captures human sampling behavior well and it reproduces a similar and heavily right-tailed distribution, as it was found in the human data. Figure 3 shows the effects of experienced variability and payoff-domain. The model correctly shows the increase in sample size due to the experienced variability. The effect of domain is less clear in the model. Whereas it correctly predicts a higher number of samples for losses than for gains if

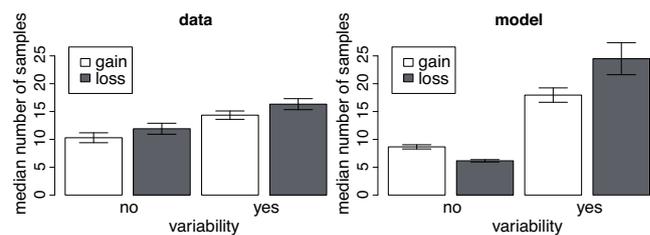


Figure 3. Sample size as a function of experienced variability and domain (shown are the average medians ± 1 SE.)

variability is experienced, it does not produce this pattern if no variability is experienced.

Discussion

How much information do people search before making a consequential choice? Our results suggest that at least two task-related factors likely affect the amount of sampled information: Sample size is likely to increase with the experienced variability of outcomes, and it will be higher if losses, rather than gains are experienced. Furthermore, our results suggest that it is possible to extend the IBL model of experiential decisions by incorporating an evidence-accumulation mechanism that predicts when people stop sampling. While the revised model presented here presents a first step into this direction, there are several potent approaches to model the accumulation process. We are currently evaluating these approaches in more detail by exploring their ability to more accurately predict human information search.

References

- Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Mahway, NJ: Erlbaum.
- Erev, I., Ert, E., Roth, A. E., Haruvy, E., Herzog, S. M., Hau, R., . . . Lebiere, C. (2010). A choice prediction competition: Choices from experience and from description. *Journal of Behavioral Decision Making, 23*, 15-47.
- Gonzalez, C., & Dutt, V. (2011). Instance-based learning: Integrating sampling and repeated decisions from experience. *Psychological Review, 118*, 523-551.
- Gonzalez, C. & Dutt, V. (2012). Transition from Exploration to Exploitation in Decisions from Sampling. Unpublished paper under review.
- Gonzalez, C., Lerch, J. F., & Lebiere, C. (2003). Instance-based learning in dynamic decision making. *Cognitive Science, 27*, 591-635.
- Hau, R., Pleskac, T. J., & Hertwig, R. (2010). Decisions from experience and statistical probabilities: Why they trigger different choices than a priori probabilities. *Journal of Behavioral Decision Making, 23*, 48-68.
- Hertwig, R., Barron, G., Weber, E. U., & Erev, I. (2004). Decisions from experience and the effect of rare events in risky choice. *Psychological Science, 15*, 534-539.
- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor, MI: University of Michigan Press.
- Ratcliff, R., & Smith, P. L. (2004). A comparison of sequential sampling models for two-choice reaction time. *Psychological Review, 111*, 333-367.

A Computational Model of Collective Sensemaking: Differential Effects of Communication Network Structure on Collective Sensemaking Abilities

Paul R. Smart (ps02v@ecs.soton.ac.uk)

School of Electronics and Computer Science, University of Southampton,
Southampton, SO17 1BJ, UK

Keywords: distributed cognition; social networks; constraint satisfaction networks; sensemaking; information sharing.

Introduction

Collective sensemaking is a form of socially-distributed cognition (see Hutchins, 1995) in which multiple agents attempt to interpret (make sense of) specific bodies of environmental information. In order to optimize performance at the collective level, agents often need to share information about the results of their own processing activity, and this raises questions about how the structure of communication networks affects collective sensemaking abilities. In the current study, we used a computational model of collective sensemaking in which individual agents are implemented as constraint satisfaction networks (CSNs) (see Smart & Shadbolt, 2012). We then investigated how the cognitive responses of agents were affected by different kinds of communication network structure.

Method

In order to explore the effect of communication network structure on the dynamics of collective sensemaking, we used a multi-agent computational model in which individual agents were implemented as CSNs. The computational architecture of the CSNs is the same as that described in Smart and Shadbolt (2012). Each agent consisted of 6 cognitive units, which represented various kinds of beliefs that agents could have about two types of object, namely cats and birds. The net activation of each cognitive unit represented the extent to which an agent held a specific belief about an object. Thus, if the net activation of the 'Cat' unit was high then the agent could be said to hold the belief represented by the 'Cat' cognitive unit. The cognitive units were connected together in such a way as to yield two kinds of interpretive response to environmental information. On the one hand, agents could interpret environmental information as indicating the presence of a cat, and, on the other hand, they could interpret environmental information as indicating the presence of a bird. Across the course of each simulation, one of these cognitive responses will tend to predominate due to the pattern of excitatory and inhibitory links between cognitive units. The way in which the activation of each cognitive unit was updated at each processing cycle is described in Smart and Shadbolt (2012).

Each of the agents within the computational model can be connected to other agents in order to create a communication network. Agents can share information about their beliefs at each cycle of a simulation in order to influence the kinds

of beliefs that their network neighbors have at the next processing cycle. The way in which the information is shared and processed by agents is described in Smart and Shadbolt (2012).

The current study examined the effect of four types of communication network structure on collective sensemaking performance. In the 'Disconnected Network' condition, all agents operated autonomously and no communication was allowed between the agents at any stage of the simulation. In the 'Random Network' condition, agents were connected together using a random network topology. The random networks, in this case, were generated following the procedure described in Mason, Jones, and Goldstone (2005). Bidirectional links between agents were added at random between the agents until a specific number of links (i.e. 1.3 times the number of agents) had been created (given that all our simulations involved 10 agents, the number of links added to random network configurations was 13 (1.3 * 10) links). In the 'Small World Network' condition, agents were connected together using a small world network topology. As with random networks, small world networks were generated using the procedure described in Mason et al. (2005). Agents were initially connected into a ring structure. Three agents were then selected at random and each of these randomly selected agents was connected to another randomly selected agent subject to the constraint that connected agents were at least 3 agents apart in the ring topology. Finally, in the case of the 'Fully Connected Network' condition, all agents were connected to all other agents using a fully-connected network topology.

Each simulation started with the creation and configuration of CSNs corresponding to individual agents. Ten agents were created for every simulation, and all agents were identical to one another in terms of their constituent architecture. Agents were then organized into one of four types of network structure as described above. It should be noted that a new network structure was created for each simulation, thus the structure of some networks (namely, the random and small world networks) was not invariant across the experimental conditions.

Table 1: Activation vectors used in the experiment.

	Fur	Meows	Cat	Feathers	Sings	Bird
Ambiguous	0.1	0.1	0.0	0.2	0.2	0.0
Unambiguous	0.5	0.5	0.0	0.0	0.0	0.0

Once the network structure had been created, the activation levels of cognitive units within each agent were initialized using one of two types of activation vector (see Table 1).

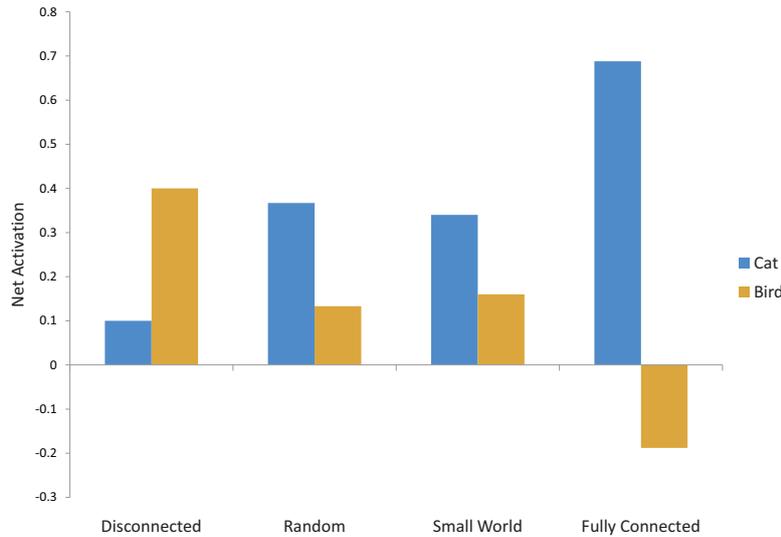


Figure 1: Mean responses of Cat and Bird cognitive units in each of the four network structure conditions.

At the start of each simulation, 4 agents were selected at random and were initialized with the ‘Unambiguous’ activation vector; the remainder of the agents were initialized with the ‘Ambiguous’ activation vector.

After the initial activation levels had been established, the simulation commenced and processing occurred in a series of processing cycles. Within each cycle, the activation of all cognitive units was updated as per the procedure described in Smart and Shadbolt (2012). The simulation continued for 20 processing cycles, and, at the end of each simulation (i.e. at the 20th processing cycle), the activation level of the ‘Cat’ and ‘Bird’ cognitive units was recorded for subsequent analysis. A total of 50 simulations were run in each of the four network structure conditions.

Results

The results are shown in Figure 1. ANOVA revealed a significant main effect of Cognitive Unit (i.e. activation of the ‘Cat’ and ‘Bird’ cognitive units) ($F_{(1,3992)} = 121.446$, $P < 0.001$) and a significant two-way interaction between the Network Structure and Cognitive Unit factors ($F_{(3,3992)} = 115.561$, $P < 0.001$). There was no significant main effect of Network Structure. Post hoc comparisons using Tukey’s HSD were performed at each level of the Cognitive Unit factor. These analyses revealed that cognitive responses in the random and small world network conditions were not significantly different from each other for either of the ‘Bird’ or ‘Cat’ cognitive units. The activation level of the ‘Cat’ cognitive unit was higher in both the random and small world network conditions as compared to the disconnected network condition, and the activation of the ‘Bird’ cognitive unit was lower in the random and small world network conditions as compared to the disconnected network condition. Activation of the ‘Cat’ cognitive unit was higher in the fully connected network as

compared to all other networks, and activation of the ‘Bird’ unit was lower in the fully connected network as compared to all other networks. Post hoc comparisons of the cognitive responses for each of the network structures revealed significant differences between the activation of ‘Cat’ and ‘Bird’ cognitive units for all network conditions. Activation of the ‘Cat’ cognitive unit was higher than ‘Bird’ cognitive units for all networks, with the exception of the disconnected network condition (see Figure 1).

Conclusion

The results of this study suggest that collective sensemaking is influenced by network structure under certain informational conditions. In all of the conditions in which agents were allowed to communicate information, a particular kind of cognitive response emerged in which cat-related beliefs predominated. This differed from the situation in which agents were not allowed to communicate (i.e. the disconnected network condition). The cognitive responses of agents that were organized into random and small world network topologies were very similar; however, they were less extreme than those of agents organized into fully-connected network topologies.

References

- Hutchins, E. (1995). *Cognition in the Wild*. Cambridge, Massachusetts, USA: MIT Press.
- Mason, W. A., Jones, A., & Goldstone, R. L. (2005). Propagation of innovations in networked groups. In *27th Annual Conference of the Cognitive Science Society*. Stresa, Italy.
- Smart, P. R., & Shadbolt, N. R. (2012). Modelling the dynamics of team sensemaking: A constraint satisfaction approach. In *Knowledge Systems for Coalition Operations (KSCO'12)*. Pensacola, Florida, USA.

Using Support Vector Machines to Determine Linear Separability

Bram Van Rensbergen (bram.vanrensbergen@ppw.kuleuven.be)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 – box 3711, B-3000 Leuven, Belgium

Eef Ameel (eef.ameel@ppw.kuleuven.be)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 – box 3711, B-3000 Leuven, Belgium

Fons Van Assche (fonzva@gmail.com)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 – box 3711, B-3000 Leuven, Belgium

Gert Storms (gert.storms@ppw.kuleuven.be)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 – box 3711, B-3000 Leuven, Belgium

Keywords: Linear separability; support vector machines; machine learning.

Introduction

Most theories on categorization agree on a main principle: category learning involves creating a category representation, and categorizing an item involves comparing that item to the representations of different categories. The theories, however, disagree on the nature of these category representations. There are two main competing lines of thought on category representations: exemplar-based theories and prototype-based theories (Valentine, 1991).

Prototype-based theories argue that objects are stored based on how similar they are to a central prototype (Rosch, 1973). In contrast, exemplar-based theories reason that objects are encoded in their absolute structure, defined by their own properties only and unrelated to any abstract summary representation (Medin & Schaffer, 1978). Years of research on the nature of categorization has resulted in mixed results, with evidence for both approaches.

One example of an issue on which exemplar- and prototype based theories make different predictions is linear separability. Two categories are considered linearly separable if a linear function of attributes exists that perfectly separates their exemplars (Ruts, Storms, & Hampton, 2004).

According to prototype-based models, for any pair of linearly separable categories represented in a geometrical space, that space is divided into two half spaces by a linear function that defines the points which are equidistant towards both prototypes. An item is then categorized in the category with the closest prototype (in that geometrical space). Thus, category membership can be determined simply by looking at the distance to the prototype, making linearly separable categories relatively easy to learn. On the other hand, categories that are not linearly separable would

take considerably longer to master, as this simple strategy of deciding on the closest prototype would not be sufficient to determine category membership.

According to exemplar-based models, on the other hand, proximity to the center of the category plays no role of any kind (Ruts, Storms, et al., 2004). Thus, exemplar-based models predict that, all other factors kept constant, linearly separable categories are not easier to master than other categories.

As prototype-based and exemplar-based models make different predictions regarding linearly separable categories, we can use those categories to shed light on the mechanisms that underlie categorization. In order to do so, however, we have to determine which categories are linearly separable, and which are not. There has been surprisingly little research into this issue. Studies that do investigate this tend to assess linear separability by first obtaining a geometric representation of the exemplars using multidimensional scaling, and then analyzing this representation with visual inspection (Malt, Sloman, Gennari, Shi, and Wang, 1999) or log linear analysis (Ruts, Storms, et al., 2004).

The current research intends to expand on these previous studies in investigating which semantic categories are linearly separable and which are not. Compared to these existing studies, however, we will use a fundamentally different technique to assess linear separability: linear support vector machines (LSVMs).

A support vector machine (SVM) is a mathematical concept used for supervised pattern learning (Vapnik, 1982; Cortes & Vapnik, 1995). Presented with a set of input data and their corresponding classes, an SVM learns which data correspond to which class. Once trained, the machine can be used for classification; for any given input, it predicts the corresponding class.

SVMs transform the input vectors into a (usually) high dimensional feature space with the help of a kernel

function, and look for the hyperplane that separates the classes optimally in that feature space (Cortes & Vapnik, 1995). SVMs put no restrictions on the nature of the kernel, allowing both linear and nonlinear functions. As the goal of our study is to assess the extent of linear separability of categories, we need a classifier that produces linear boundaries; as such, we can only apply linear kernel functions.

Compared to assessing linear separability by analyzing a geometric representation obtained with multidimensional scaling, LSVMs hold several advantages. For one, there is no issue of choosing the optimal dimensionality, as LSVMs always use the maximal number of dimensions present in the data. Secondly, LSVMs make no assumptions about the nature of the distribution of the items. This is in contrast to many statistical criteria used to analyze the geometric representation obtained with multidimensional scaling, which do put restrictions on the distribution.

Method and Results

In a first study, we examined the linear separability of natural and artifact concepts. The idea was to teach a LSVM to use feature values to predict category membership, and then to examine which categories the LSVM could linearly separate from one another. We looked at six pairs of natural categories and five pairs of artifact categories. Each pair consisted of 61 to 85 exemplars, which were rated on 30 to 51 features. We found that LSVMs are very efficient at using feature values to predict to which class an item belongs. Prediction accuracy was high both for natural classes (up to 100% accuracy) and for artifact classes (up to 97.07% accuracy). Additionally, we found that some of the natural categories were linearly separable and some were not, and that none of the artifact categories could be considered linearly separable.

A second study again examined the linear separability of natural and artifact concepts, but this time in a multiclass environment. We made use of two datasets, one comprising 129 animals divided over five natural categories, described by 764 features, and the other containing 166 artifacts divided over six artifact categories, described by 1295 features. We found that multiclass LSVMs could efficiently use these feature values to predict category membership: Prediction accuracy was high for both natural classes (up to 98.78% accuracy) and artifact classes (up to 99.29% accuracy). We found that all natural categories were linearly separable from one another, except for the fish and mammal categories, and that most of the artifact categories were linearly separable from one another as well.

In our third study, we investigated whether LSVMs can use similarity ratings to linearly separate the different names people give to types of movement. We examined two datasets: one with data from English-speaking students, and one with data from Dutch speaking students, each containing the dominant name and similarity ratings for 24 video clips depicting movement. We found that LSVMs

could use similarity ratings to predict the name participants give to a type of movement, with a maximal predictive accuracy of 95.25% for the English dataset, and 79.7% for the Dutch dataset. Additionally, we found that for both datasets, some of the categories were sufficiently linearly separable from one another, and some were not.

Conclusion

We demonstrated that linear support vector machines can be used efficiently to determine the relative linear separability of semantic concepts. We showed how LSVMs can use feature values or similarity ratings to predict category membership, and how we can use the LSVMs misclassifications to determine the extent of the linear separability of the tested categories.

Acknowledgments

The first author was funded by research grant OT/10/024.

References

- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273-297.
- Malt, B. C., Sloman, S. A., Gennari, S., Shi, M., & Wang, Y. (1999). Knowing versus naming: Similarity and the linguistic categorization of artifacts. *Journal of Memory and Language, 40*(2), 230-262.
- Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review, 85*(3), 207-238.
- Rosch, E. (1973). Natural categories. *Cognitive Psychology, 4*, 328-350.
- Ruts, W., Storms, G., & Hampton, J. (2004). Linear separability in superordinate natural language concepts. *Memory & Cognition, 32*(1), 83-95.
- Valentine, T. (1991). A unified account of the effects of distinctiveness, inversion, and race in face recognition. *The Quarterly Journal of Experimental Psychology, 43*(2), 161-204.
- Vapnik, V. (1982). *Estimation of dependences based on empirical data*. New York: Springer-Verlag.

A Rational Analysis of Memory Search Termination

Eddy J. Davelaar (eddy.davelaar@gmail.com)

Department of Psychological Sciences, Birkbeck University of London
Malet Street, WC1E 7HX, London, UK

Erica C. Yu (ericayu@umd.edu)

J. Isaiah Harbison (isaiah.harbison@gmail.com)

Erika K. Hussey (erikahussey@gmail.com)

Michael R. Dougherty (mdougher@umd.edu)

Department of Psychology, University of Maryland at College Park
College Park, MD 20742 USA

Abstract

An important component of many, if not all, real-world retrieval tasks is the decision to terminate memory search. Despite its importance, systematic evaluations of the potential rules for terminating search are scarce. Recent work has focused on two variables: the total time spent in memory search before search is terminated and the exit latency (the time between the last retrieved item and the time of search termination). These variables have been shown to limit the number of plausible rules for terminating memory search. Here, we derive a closed-form expression of the exit latency function using a rational approach. We show its goodness of fit to existing data and make testable predictions about the influence of changing the relative costs and benefits of memory search. Results from an experiment are presented that support the model's predictions. We conclude that the decision to terminate memory search is based on moment-to-moment changes in subjective utility of retrieved memories.

Keywords: Stopping rule; memory retrieval; free recall.

Introduction

One of the most influential developments in cognitive psychology and cognitive science is that of a detailed theoretical framework of memory processes. In the late 1960s, Murdock (1967) summarized a view held by many theorists into the "modal model", a model in which information (memoranda) transfers from sensory memory to short-term memory and then to long-term memory, with each subsequent system having greater memory persistence. The modal model was mainly a framework of memory encoding and the details of memory retrieval were left less-specified. Later theories explicated the retrieval processes in more detail (Anderson, 1972; Metcalfe & Murdock, 1981; Raaijmakers & Shiffrin, 1981). A common aspect in these theories is the assumption that retrieval from memory can be seen as a search process (Yntema & Trask, 1963) which takes time to complete. Importantly, in order to characterize this search process, models of memory were endowed with stopping rules that prevent the models from continuing search indefinitely. Despite the fact that theoreticians have been quick to incorporate stopping rules into models of memory, research evaluating the class of stopping rules that

might characterize people's decision to terminate memory search is limited.

The evaluation of stopping rules in models of recall is of both theoretical and practical interest. From a theoretical perspective, the goal of developing a comprehensive model of memory retrieval necessitates that we specify the control systems that operate on the memory representations (Newell, 1973). Any particular memory model might yield qualitatively different predictions depending on the specification of the control structures. This is particularly true for stopping rules, since the particular stopping rule employed will affect how long the model will persist in search, which can potentially affect the output of the model (number of items retrieved) and retrieval latencies.

From a practical perspective, understanding stopping rules in the domain of memory retrieval can be informative for the development of artificial intelligence and decision support systems, as well as for cognitive models of diagnostic hypothesis generation and judgment (Thomas, Dougherty, Sprenger, & Harbison, 2008). Within these systems, different stopping rules may yield qualitatively different solutions to diagnostic problems.

In this paper, we extend the analytical work by Harbison et al. (2009) and derive a stopping rule motivated by a rational analysis of memory (Anderson & Milson, 1989). The resulting closed-form expressions are tested against existing and new data.

Stopping rules

Atkinson and Shiffrin (1968, page 121) suggested a number of stopping rules, which have been implemented in models by a number of authors. These different stopping rules are: an internal time limit (Davelaar, et al, 2005; Davelaar, 2007; Diller, Nobel & Shiffrin, 2001; Farrell & Lewandowsky, 2002; Metcalfe & Murdock, 1981), a strength threshold (Anderson, et al. 1998; Diller, Nobel & Shiffrin, 2001), and an event-counter that would terminate search after a prespecified number of events (Raaijmakers & Shiffrin, 1981; Shiffrin, 1970).

Given the various stopping rules employed in the literature, it is clear that little heed has been paid to how a chosen stopping rule might affect the model's retrieval

dynamics. Furthermore, the empirical research on which to test candidate stopping rules has been missing. The presence of self-terminating stopping rules in models of memory is in recognition of the fact that human observers are often required to self-terminate retrieval. Yet, most empirical studies of free recall have masked the contribution of stopping rules by providing participants with a pre-set retrieval interval. The use of pre-set retrieval intervals eliminates the need for the participant to utilize a stopping rule and even if participants were to use such a rule there would be no method of measuring it.

In order to address stopping rules in recall, one needs to allow participants to terminate their own retrieval episode. Consequently, the procedure of interest here is one in which the participant is given potentially infinite amount of time for retrieval, but allowed to terminate retrieval whenever they wish (Dougherty & Harbison, 2007; Harbison, et al., 2009). This paradigm yields two temporal variables anticipated by models of memory that are important for understanding search termination, but which have received relatively little attention in the literature. The first of these response time measures is total time. Total time indexes the elapsed time between the onset of a retrieval cue (i.e., the initiation of the retrieval episode) and the decision to terminate retrieval (i.e., termination of the retrieval episode). The fact that models of memory incorporate stopping rules suggests that these models yield total time predictions. Obviously, different stopping rules will yield different total time predictions, but on an intuitive level one would expect total time to be monotonically related to total number of items retrieved: Total time should increase with the number of items retrieved.

The second response time measure is what Dougherty and Harbison (2007) called the exit latency. Exit latencies index the amount of time between the final successful retrieval and the decision to terminate of search. In contrast to total time, there is no obvious, intuitive prediction regarding how long participants will persist in retrieval as a function of number of successful retrieval attempts. Thus, exit latencies provide a potentially diagnostic source of data for evaluating stopping rules, particularly when considered in conjunction with the total time measure.

Few published studies report data on the two temporal variables relevant for measuring termination decisions (Dougherty & Harbison, 2007; Harbison, et al., 2009; Unsworth, Brewer & Spiller, 2011). In the study by Dougherty and Harbison (2007), participants were visually presented with a cue word and 10 target words (A-X₁, A-X₂, ..., A-X₁₀). They were told to remember the target words that were presented with each cue word. Each list of 10 target words had a unique cue word. Twelve such lists were presented in blocks of three. After each block of lists were presented, participants were given a cue word and had to report verbally as many words studied with that cue word (A-?) as they could. Responses were recorded and participants pressed the space-bar to indicate that they could not generate additional words. The total time participants

spent in search was measured as the time between presentation onset of the cue for retrieval and the time of pressing the space-bar. The exit latency was measured as the time interval between the last retrieved item and the time of pressing the space-bar.

Figure 1 presents the pattern of results regarding the stopping and exit latencies as a function of the number of words retrieved in that trial. The solid lines are the best-fitting curves from the rational analysis introduced later. Figure 1 shows that total time is an increasing function of the number of words retrieved in that trial, whereas that exit latency is a negatively decelerating function of the number of words retrieved in that trial.

Evaluating Stopping Rules

Harbison et al. (2009) conducted a simulations study to compare several of stopping rules suggested by Atkinson and Shiffrin (1968). Of these rules, only the total number of failures rule fitted the data both qualitatively and quantitatively. The total number of failures rule is a special case of an iterative rule that is only concerned with the current sample from memory and the total accumulated number of failures. This lends itself to a rational analysis of the same rule which can make novel predictions.

We see memory retrieval as a form of information sampling for which a cost is incurred with every sampling attempt and a benefit is obtained for successful retrievals. We define the memory value function in which the total net value during the retrieval phase is a function of the total number of items retrieved at the elapsed retrieval time.

We set out to derive a closed-form expression for the exit latency, where the decision to terminate search depends only on the information of the last time-step. We converged on the following rule:

Terminate search when the additional cost of retrieving the next item starts to outweigh the relative benefit of having retrieved that item (akin to a Weber fraction).

The derivation of the closed-form expressions is based on a value function. We assume that a cost, a , is incurred with every sampling attempt, t , and a benefit, b , is obtained with every successful retrieval. This assumption mirrors the rational analyses of memory (Anderson & Milson, 1989) and problem solving (Payne & Duggan, 2011), which are in turn grounded in the animal foraging theory. We define the memory value function as:

$$V_t = Q + bN(t) - at \quad (1)$$

where b and a are the benefit and cost parameters. $N(t)$ is the total number of items retrieved at time t . Here we use:

$$N(t) = L(1 - e^{-\lambda(t-\delta)}) \quad (2)$$

with listlength, L , rate of cumulative retrieval, λ , and offset for starting retrieval, δ . This equation has been shown to

provide good approximations to observed data (Wixted & Rohrer, 1994) and follows from a sampling-with-replacement model (Indow & Togano, 1970). To compare, a sampling-without-replacement model would retrieve all items in a free recall task irrespective of the number of items to be retrieved. This counters actual observed data. The net_value, V_t , has a constant, Q , which is a (currently unconstrained) free parameter that allows the initial Q/a seconds to be without retrieval and is therefore related to the maximum first recall latency.

It is assumed that the participant aims to obtain the maximal possible net_value. That is the participant stops search when V_t is maximal, i.e., $dV_t/dN(t) = 0$.

$$dV_t/dt = bL\lambda e^{-\lambda(t-\delta)} - a$$

$$t_{\text{stop}} = -\lambda^{-1} \ln(a/bL\lambda) + \delta \quad (3)$$

Substituting t_{stop} for t in (2) and solving for N_{stop} gives:

$$N_{\text{stop}} = L - a/b\lambda$$

The derivation of the exit latency function is based on the additional cost of retrieving the next item compared to the relative benefit of having retrieved that item:

$$\text{cost}(t + \Delta t) - \text{cost}(t) = b/V_t$$

$$a(t + \Delta t) - at = b/(Q + bN_{\text{stop}} - at_{\text{stop}})$$

$$t_{\text{exit}} = \frac{b}{a(Q + bN_{\text{stop}} - at_{\text{stop}})}$$

$$t_{\text{exit}} = \frac{b}{a(Q + bN_{\text{stop}} + a(\frac{1}{\lambda} \ln(1 - \frac{N_{\text{stop}}}{L}) - \delta))}$$

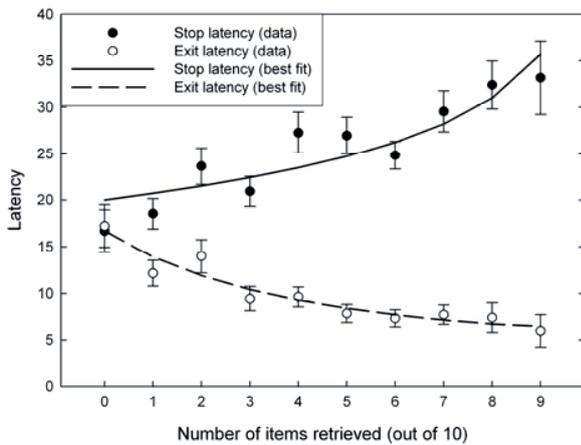


Figure 1. Empirical data from Dougherty and Harbison (2007) with fits of the best fitting model ($R^2 = .946$). Error bars represent the standard error of the means.

Figure 1 shows the fits of this model to the data by Harbison et al. (2009). The model predicts that when the cost increases (or benefit decreases) the search with terminate sooner. These predictions are tested in the experiment described next.

Experiment

Methods

Participants

Forty-five participants were recruited from the University of Maryland subject pool and received performance-based compensation (\$15 or \$20) for participation in the study. Two participants were removed from analysis due to data collection errors.

Design and materials

The design used a delayed free recall paradigm whereby participants studied word lists, completed distractor math problems, and verbally recalled words from the most recent list using a PC-based microphone. The session was presented in two blocks. The first was a baseline block of 16 trials with the same payoff structure across participants (+100 for a correct recall, -100 for each second spent and incorrect recall). In the second block, cost and reward were varied between participants: one group was given an increase in reward (+150) for a correct recall and a simultaneous decrease (-50) in each second spent and each incorrect recall; the other group was given the inverse (+50 rewards, -150 cost). The retrieval protocol followed the self-terminated search paradigm used by Dougherty and Harbison (2007): participants were instructed that they had unlimited time to recall words and could end the recall period at any time by pressing the spacebar. The experimenter monitored the participant's recall and updated the participant's score in real-time, providing feedback to the participant on screen. Thirty-two lists of monosyllabic words were randomly created for each participant. List length was varied between 5, 7, 9, and 11 words and presentation order was randomized to prevent strategy use.

Procedure

Participants were informed they would complete a verbal recall task. The study words were sequentially presented in the center of the computer monitor for 2 s each. Following each study list, a distractor task was presented, which consisted of two simple, timed math problems. Problems contained three digits and two operands (e.g., $3 * 2 + 1 = ?$) and always resulted in a single-digit answer (digits 0-9). A question mark prompted the participant to enter an answer. Components of the math problem were presented sequentially for 1 s each. After two math problems, participants were prompted to begin verbally recalling words from the most recent study list and press the spacebar when they were finished retrieving. After the spacebar press, participants were prompted to press the spacebar again to begin the next study list when they were ready.

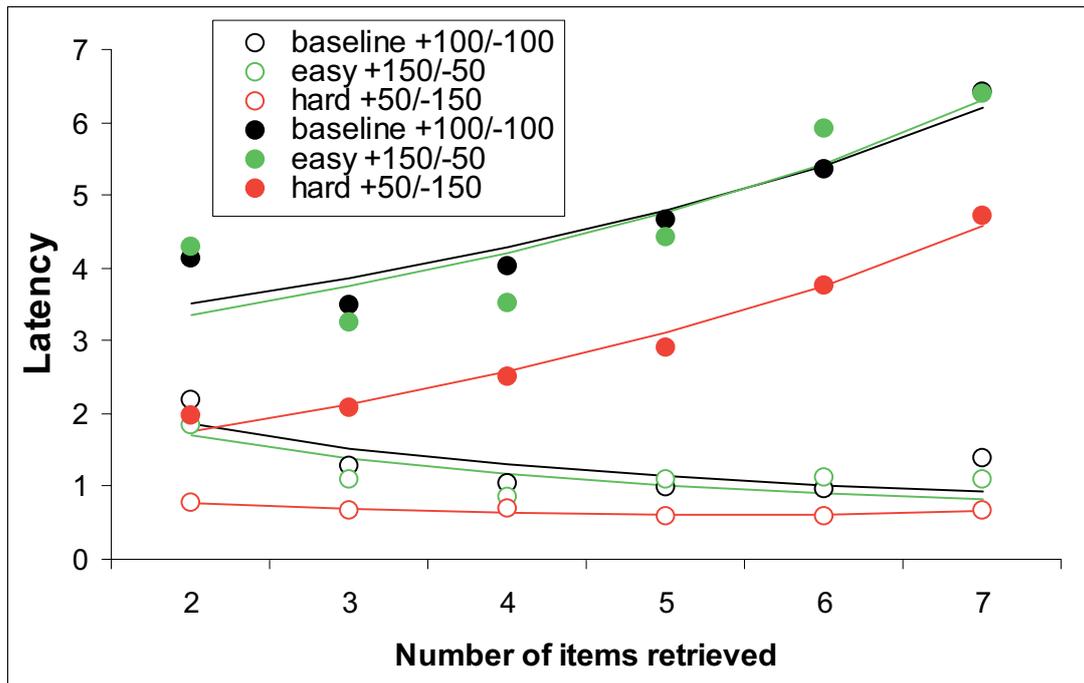


Figure 2. Total time and exit latency functions for the baseline block (both groups combined) and the second block (easy and hard condition). Only the last 8 trials of each block were used. Lines represent the best-fitting model ($R^2 = .97$)

Coding

Using PennTotalRecall audio-analysis software, verbal retrieval data were retrospectively analyzed with millisecond accuracy. Two coders independently coded: 1) all words that were produced by each participant on each trial, 2) the time stamps of the verbal onset of all generated words, and 3) the time stamps of retrieval termination (i.e., times associated with spacebar presses). From these data, number of items retrieved, number of intrusions including repetitions and intra- and extra-list false alarms, inter-retrieval times, and exit latencies (i.e., time between end of final word retrieved and retrieval termination) were calculated. Each subject's trials were averaged before summarizing across subjects.

Results

A 2x2 mixed design included an initial baseline control environment (+100 correct recall, -100 second spent or incorrect recall) and a second payoff environment varied between subjects (easy: +150, -50; hard: +50, -150). Due to steep learning curves in each new environment, only the last 8 of the 16 trials in each block were included in the following repeated measures ANOVA analyses.

The net points (rewards for correct recalls less the penalties for incorrect recalls and time spent) were updated in real-time for participants to use as feedback to monitor their own retrieval performance. As predicted, net points earned in each block increased over time [$F(1,41) = 6.772$, $p < .013$, $\eta_p^2 = .142$] and the participants for whom the rewards increased and costs decreased earned more points

overall [$F(1,41) = 15.230$, $p < .001$, $\eta_p^2 = .271$]; while net points in the baseline block were equivalent across conditions (easy: $M = -23.214$, $SE = 41.040$; hard: $M = -35.795$, $SE = 40.097$), performance splits drastically in the second block (easy: $M = 281.845$, $SE = 54.965$; hard: $M = 161.080$, $SE = 53.706$; condition x time: 38.803 , $p < .001$, $\eta_p^2 = .486$).

Total number recalled, including intrusions and repetitions, did not vary due to time, payoff environment, or an interaction of the two [conditions: $F(1,41) = 1.611$, ns, $\eta_p^2 = .038$; time: $F(1,41) = 3.361$, ns, $\eta_p^2 = .076$; condition x time: $F(1,41) = 3.843$, ns, $\eta_p^2 = .086$].

Temporal measures were sensitive to learning across the experiment: total time and exit latency both improved significantly for all participants [total time: $F(1, 41) = 22.186$, $p < .001$, $\eta_p^2 = .351$; exit latency: $F(1,41) = 12.95$, $p < .001$, $\eta_p^2 = .240$]. This performance improvement came primarily from the participants for whom the rewards decreased and the costs increased: the interaction between time and payoff structure was significant for both measures [total time: $F(1,41) = 29.009$, $p < .001$, $\eta_p^2 = .414$; exit latency: $F(1,41) = 9.982$, $p < .003$, $\eta_p^2 = .196$], but the main effects of condition were not significant [total time: $F(1,41) = 1.138$, ns, $\eta_p^2 = .027$; exit latency: $F(1,41) = 2.537$, ns, $\eta_p^2 = .058$].

Figure 2 shows the data on the retrieval latencies broken down by block and condition. The solid lines are the best fits of the model. The prediction was that increase in cost or decrease in benefit would lower the latencies. Compared to the baseline condition, making the test harder by increasing the cost and decreasing the benefit did indeed lower all

retrieval latencies. Nevertheless, the opposite manipulation, decreasing the cost while simultaneously increasing the benefit, did not change the latencies compared to baseline. We address this asymmetry in the general discussion.

General Discussion

The purpose of this paper was to derive and test a stopping mechanism that was motivated by a rational analysis of decision made with every memory sample. The resulting closed-form expression of the exit latency function fits the data presented by Dougherty and Harbison (2007) and makes testable predictions about the influence of monetary payoff structure on retrieval latencies and the decision to stop memory search.

The prediction was that making it harder to gain points would lower the retrieval latencies due to higher probability of stopping, whereas the reversed would be the case when it was easier to gain points. Interestingly, only the former prediction was borne out by the data and model fits. The results might be seen as an instance of loss aversion by suggesting what could be called an “it-ain’t-broke” hypothesis. Loosely put, when it is harder to obtain points, the cognitive system readjusts itself to avoid losing too much. However, when the environment changes to such an extent that it becomes easier to gain points, the system will not calibrate itself to then minimize the gains. Hence, if the cognitive system is not losing by what it does (i.e., it-ain’t-broke) then there is no reason for adjusting the cognitive parameter (i.e., don’t-fix-it).

Anderson and colleagues provided a rational analysis of the free recall task (Anderson & Milson, 1989; Anderson & Schooler, 1991), in which each item has a need probability associated with it. Only those items are retrieved whose need probability is larger than a certain criterion, which increases with the time spent inspecting an item before accepting or rejecting it. Anderson and Milson (1989) were able to capture a number of basic memory phenomena using their adaptive perspective. However, their analysis only provided the time of the last retrieved item and not of the exact time of terminating memory search. A possibility would be to use the criterion to estimate the termination time, but this would require knowing the functional form of how the criterion changes during item inspection. Nevertheless, the success of Anderson’s rational analysis and our current results warrants investigating how these can be combined and would allow analyzing the consequences of different retrieval processes on stopping rules. We leave such an endeavor for the future.

Our analysis suggests that stopping rules should play a more central role in the development and testing of models of memory. The choice of stopping rule has major impact on the overall model behavior. Obviously, one of the ultimate goals of memory theory is to characterize memory retrieval in general, both in and out of the lab. By focusing more on how people terminate memory search, we can bring our models more in line with the type of retrieval tasks that

characterize retrieval tasks outside of the free-recall paradigm.

Future work could address a number of remaining issues regarding the model. First, in our experiment we treated a single intrusion as equally costly as one second in the retrieval phase. It may require more or less than one second to utter an intrusion depending on word characteristics such as word frequency and word length. Although intrusions tend to be rare in free recall, assessing the cost of an intrusion would be one area of further inquiry. Second, and related to the first, Laming (2009) suggested that free recall terminates when the same word gets retrieved. However, an implementation of this rule did not produce the exit latency curve seen in the data. A formal model comparison could elucidate the status of a repetition as either an initiator of search terminations or a covariate. Third, in our subjective value function, the value of Q is arbitrary. In our current conception, we think that Q might be related to such factors as motivation to retrieve or time pressure. Finally, in the area of task interleaving, Payne, Duggan and Neth (2011) addressed stopping behavior on easy and difficult tasks. A formal model was developed based on foraging theory, which fitted the data remarkably well. Our rational analysis could be compared to stopping rules based on foraging theory (cf. Wilke, Hutchison, Todd & Czienskowski, 2009).

Investigating stopping rules has important implications for understanding tasks other than free-recall. For example, within the medical decision making literature, it is clear that physicians entertain costs when determining when to terminate their retrieval of diagnostic hypotheses from memory (Weber et al., 1993). More recently, Dougherty and Hunter (2003a; 2003b) showed that the perceived probability of any particular event (a hypothesis) is partially dependent on the number of alternatives retrieved from memory, which was affected by time pressure. This suggests that when one decides to terminate memory search will affect his or her perceived probability of a particular hypothesis. Within the frequency judgment literature, Brown and colleagues (Brown, 1995; 1997; Brown & Sinclair, 1999; Conrad, Brown, & Cashman, 1998) have shown that participants’ responses to survey questions often are a monotonically increasing function of total time spent retrieving searching memory. Thus, the magnitude of participants’ frequency judgments on behavioral survey questionnaires should be affected by when they terminate search of long-term memory. Although the above tasks are all quite distinct, they serve to underscore the ubiquity of stopping rules in real-world retrieval tasks. Therefore, understanding how people terminate memory search, and the psychological variables that affect search termination, is paramount to the development of comprehensive models of memory retrieval and to understanding the dynamics of memory retrieval outside the lab.

In summary, in this paper we obtained further evidence for the view that participants are making adaptive choices to search termination that are based on a cost-benefit analysis.

Acknowledgments

Part of this material is based on work supported by the National Science Foundation under Grants SES-0134678 and BCS-1030831. The authors thank Dave Huber and Rick Thomas for their helpful comments.

References

- Anderson, J. R. (1972). FRAN: a simulation model of free recall. In G. H. Bower, & J. T. Spence (Eds.), *The psychology of learning and motivation: Vol. 5. Advances in research and theory* (pp. 315-378). New York: Academic Press.
- Anderson, J. R., & Milson, R. (1989). Human memory: an adaptive perspective. *Psychological Review*, *96*, 703-719.
- Anderson, J. R., & Schooler, L. J. (1991). Reflections of the environment in memory. *Psychological Science*, *2*, 396-408.
- Anderson, J. R., Bothell, D., Lebiere, C., & Matessa, M. (1998). An integrated theory of list memory. *Journal of Memory and Language*, *38*, 341-380.
- Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: a proposed system and its control processes. In K. W. Spence, & J. T. Spence (Eds.), *The psychology of learning and motivation Vol. 2. Advances in research and theory* (pp. 89-195). New York: Academic Press.
- Brown, N. R. & Sinclair, R. C. (1999). Estimating number of lifetime sexual partners: Men and women do it differently. *Journal of Sex Research*, *36*, 292-297.
- Brown, N. R. (1995). Estimation strategies and the judgment of event frequency. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *21*, 1539-1553.
- Brown, N. R. (1997). Context memory and the selection of frequency estimation strategies. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *23*, 898-914.
- Conrad, F., Brown, N. R., & Cashman, E. (1998). Strategies for estimating behavioral frequency in survey interviews. *Memory*, *6*, 339-366.
- Davelaar, E. J. (2007). Sequential retrieval and inhibition of parallel (re)activated representations: a neurocomputational comparison of competitive queuing and resampling models. *Adaptive Behavior*, *15*, 51-71.
- Davelaar, E. J., Goshen-Gottstein, Y., Ashkenazi, A., Haarmann, H. J., & Usher, M. (2005). The demise of short-term memory revisited: empirical and computational investigations of recency effects. *Psychological Review*, *112*, 3-42.
- Diller, D. E., Nobel, P. A., & Shiffrin, R. M. (2001). An ARC-REM model for accuracy and response time in recognition and recall. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *27*, 414-435.
- Dougherty, M. R. P., & Hunter, J. E. (2003a). Hypothesis generation, probability judgment, and individual differences in working memory capacity. *Acta Psychologica*, *113*, 263 – 282.
- Dougherty, M. R. P., & Hunter, J. E. (2003b). Probability judgment and subadditivity: The role of working memory capacity and constraining retrieval. *Memory & Cognition*, *31*, 968-982.
- Dougherty, M. R., & Harbison, J. I. (2007). Motivated to retrieve: how often are you willing to go back to the well when the well is dry? *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *33*, 1108-1117.
- Farrell, S., & Lewandowsky, S. (2002). An endogenous distributed model of ordering in serial recall. *Psychonomic Bulletin & Review*, *9*, 59-79.
- Harbison, J. I., Dougherty, M. R., Davelaar, E. J., & Fayyad, B. (2009). On the lawfulness of the decision to terminate memory search. *Cognition*, *111*, 146-421.
- Indow, T., & Togano, K. (1970). On retrieving sequence from long-term memory. *Psychological Review*, *77*, 317-331.
- Laming, D. (2009). Failure to recall. *Psychological Review*, *116*, 157-186.
- Metcalfe, J., & Murdock, B. B. (1981). An encoding and retrieval model of single-trial free recall. *Journal of Verbal Learning and Verbal Behavior*, *20*, 161-189.
- Murdock, B. B. (1967). Recent developments in short-term memory. *British Journal of Psychology*, *58*, 421-433.
- Newell, A. (1973). You can't play 20 questions with nature and win: Projective comments on the papers of this symposium. In W. G. Chase (ed.), *Visual Information Processing*. New York: Academic Press.
- Payne, S. J., & Duggan, G. B. (2011). Giving up problem solving. *Memory & Cognition*, *39*, 902-913.
- Payne, S. J., Duggan, G. B., & Neth, H. (2007). Discretionary task interleaving: heuristics for time allocation in cognitive foraging. *Journal of Experimental Psychology: General*, *136*, 370-388.
- Raaijmakers, J. G. W., & Shiffrin, R. M. (1981). Search of associative memory. *Psychological Review*, *88*, 93-134.
- Shiffrin, R. M. (1970). Memory search. In D. A. Norman (Ed.), *Models of human memory*. New York: Academic Press.
- Thomas, R. P., Dougherty, M. R., Sprenger, A. M., & Harbison, J. I. (2008). Diagnostic hypothesis generation and human judgment. *Psychol. Review*, *115*, 155-185.
- Unsworth, N., Brewer, G. A., & Spillers, G. J. (2011). Factors that influence search termination decision in free recall: an examination of response type and confidence. *Acta Psychologica*, *138*, 19-29.
- Wilke, A., Hutchinson, J. M. C., Todd, P. M., & Czienskowski, U. (2009). Fishing for the right words: decision rules for human foraging behavior in internal search tasks. *Cognitive Science*, *33*, 497-529.
- Wixted, J. T., & Rohrer, D. (1994). Analyzing the dynamics of free recall: an integrative review of the empirical literature. *Psychonomic Bulletin & Review*, *1*, 89-106.
- Yntema, D. B., & Trask, F. P. (1963). Recall as a search process. *Journal of Verbal Learning and Verbal Behavior*, *2*, 65-74.

Modeling representational shifts in learning the number line

Leif K. Lende (leif.k.lende@diakonhjemmet.no)
Department of Social Education and Occupational Therapy,
Diakonhjemmet University College Rogaland
Vaagsgaten 40, N-4306 Sandnes, Norway

Niels Taatgen (n.a.taatgen@rug.nl)
Department of Artificial Intelligence,
University of Groningen
Nijenborgh 9, 9747 AG Groningen, Netherlands

Abstract

On the basis of findings from an experiment with 6-year-old children we show a proposal for a cognitive model of representational shifts in learning the number line. The findings from the experiment provide information on number line estimation - that is, translating a number to a spatial position on a number line. Though the experiment is a replication of an experiment done by Siegler and Ramani (2008) where they concluded with a logarithmic to linear shift, we could not find logarithmic representation of the results from any of our subjects. What we find is anchor points as important for improvement on learning the number line.

Keywords: Learning; numerical magnitudes; number line; dynamic decision making; memory; cognitive architectures; ACT-R.

Introduction

In this paper we present a model of the learning process involved when dealing with the estimation of what position a number value has on a number line.

The learning sequence involved is the one that Siegler calls the logarithmic-to-linear shift in representations of numerical magnitude (Siegler, Thompson, & Opfer, 2009).

Siegler et al (2009) show that children undergo parallel changes from logarithmic to linear representation on numerosity estimation tasks.

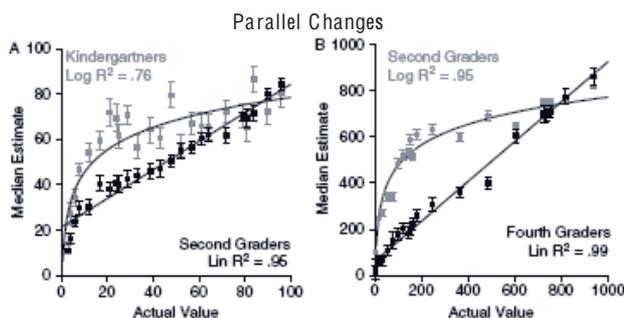


Figure 1. The logarithmic to linear shift. From Siegler, Thompson, & Opfer. (2009), Copyright 2009 Wiley. Reprinted with permission.

The example we have reused from their article in figure 1 shows long-term changes in estimation of whole number magnitudes. (A) On 0–100 number lines, kindergartners' estimates were better fit by the logarithmic function than by the linear, whereas second-graders' estimates were better fit by the linear function than by the logarithmic; (B)

On 0–1000 number lines, second-graders' estimates were better fit by the logarithmic function than by the linear, whereas fourth-graders' estimates were better fit by the linear function than by the logarithmic.

The explanation by Siegler et al. was challenged by others (Barth & Paladino, 2011). They point out that one of the challenges of putting a number on the number line is to have a sense of proportion: what exactly is the length of a single unit? This is not a trivial question for children that do not yet have a sense of what division is.

Our own earlier work also showed that a simple Weber explanation of the learning sequence of the logarithmic to linear shift does not hold as a complete explanation (Lende & Taatgen, 2011). We proposed that a possible account for the transition towards a linear representation is that children learn the location of particular points on the number line. Schneider et al. (2008) showed that the distribution of fixations on the number line for all three groups of first grade, second grade and third grade children are concentrated around beginning, midpoint and ending of the number line, suggesting that at least these three points are represented separately (Figure 2).

Distribution of fixations

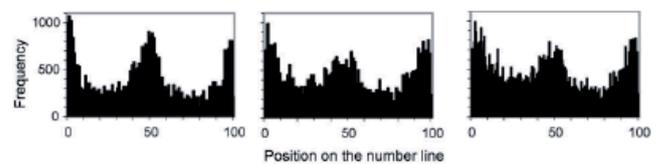


Figure 2: Distribution of fixations on the number line (left: first grade; middle: second grade; right: third grade). From Schneider et al. (2008), Copyright 2008 Elsevier. Reprinted with permission.

In addition, their work shows that from grade 1 to 3 children tend to increasingly focus on the correct positions on the number line while solving the estimation tasks.

Because of the mentioned challenges to the explanation of Siegler et al. and that it is hard to see from aggregated data what is going on with individuals; we have designed our experiment as a replication of Siegler and Ramani (2008) with the goal to look at individuals and the goal to build a model.

The number line estimation task

The experiment is a replication of a study by Siegler and Ramani (2008) among preschool children from low

income families. Siegler and Ramani found a striking improvement in number-line performance in the children after they had played a board game involving counting, but not on a board game involving colors.

The Outline of the experiment

The experiment consisted of four elements: a pretest, a training program of two weeks, a posttest and finally a second post test to measure long-term learning. We will not discuss the results of the second post test here.

After the pretest, the sample group was provided with the same training program as Ramani and Siegler used for their test of preschoolers (Siegler & Ramani, 2008). Children met one-on-one with an experimenter for four 15-minute sessions within a 2-week period. After the 2 weeks the first posttest was conducted. Then after seven new weeks a second posttest was conducted. All tests are the same.

The Method of the three tests

Participants

Participants were 39 Norwegian children in their first year of school, so-called preschool, with no experience with number lines. All of them are born in 2004 and recruited from the same municipality, Gjesdal. 17 of them are recruited from Solås School, 7 from Dirdal School and 15 from Bærland School. The population at these schools is mixed, but at Bærland with a larger representation of bilingual children, Norwegian not being their mother tongue. 21 of the participants participated in the experiment while the rest of them acted as a control group.

Materials

Stimuli for the number line estimation task were two stacks of 10 sheets of paper, each with a 25 cm long line arranged horizontally across the page, with '0' just below the left end of the line, and '10' just below the right end. A number from 1 to 10 inclusive was printed approximately 3 cm above the center of the line, with each number printed on one of the 10 sheets in each stack. The order of the sheets in the stack was randomized.

Procedure

The test is conducted as a teacher to student task:

- The teacher or student pulls a sheet from the stack.
- The teacher says: "Here is the number [number that is on the pulled sheet]. And here you see a line that starts with 0 and ends at 10. Where on this line is the correct position for the number you see? Put a mark with your pencil".
- The student makes a mark where he or she thinks the number should be positioned. There is no time constrain for the subject to fulfill the task.

The task is carried out with all the sheets in the first stack. Then the task is continued in the same way with the second stack. In this way the numbers from 1 to 10 inclusive were presented twice in random order, with all

numbers presented once before any number was presented twice. No feedback was given, only general praise and encouragement.

Method of the Board Game

In the training program between the pretest and the first posttest the subjects played a board game using a play button to move along a line of squares from square to square.

Materials

The board game for the experiment group shown in figure 3 consists of a number line with numbers in colored squares from 1 to 10 with a blank square as starting position for the game.

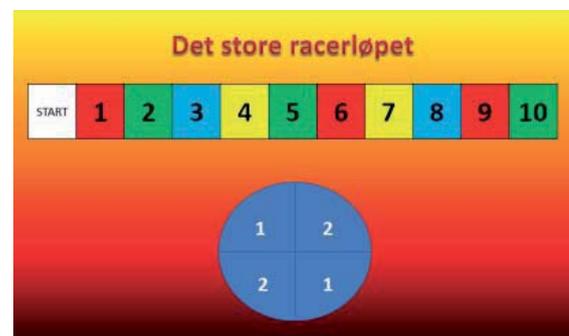


Figure 3. The Game board for the experiment group

Beneath the number line there is a circle with a spinner. (The spinner is not shown on the figure) In each quarter of the circle the numbers one or two is printed.

The board game for the control group consists of a similar line of squares, but with no numbers as shown in figure 4.

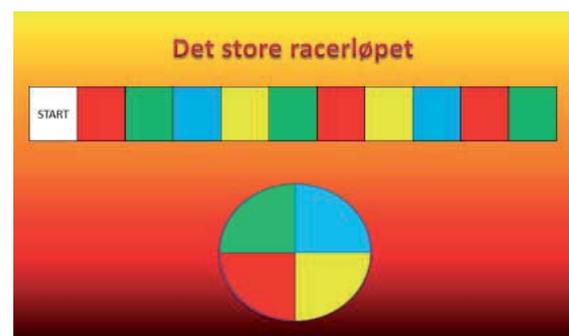


Figure 4. The Game board for the control group

Beneath the line of colored squares there is a circle with a spinner. (The spinner is not shown on the figure) Each quarter of the circle is painted with different colors corresponding to the colors used in the line of squares.

Procedure

The subjects trained with their board games for 15 minutes twice a week for 2 weeks.

When a subject of the experiment group turns the spinner the player moves his play button as many squares as the spinner tells (1 or 2 steps) while saying out loud the numbers in the squares he steps on.

When a subject of the control group turns the spinner the player moves his play button to the first square on the line of squares that is painted with the same color as given by the spinner.

Result and discussion

Figure 5 and 6 show the mapping between numbers and positions on the number line that we found in the pretest and the first posttest of the experiment. Performance is on average reasonably good.

It is surprising that where the curve differs from linear, it is not towards a logarithmic curve, but in the opposite direction..

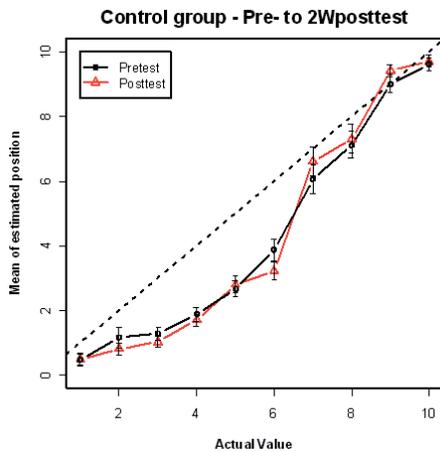


Figure 5. The figure shows the average result of the control group positioning the numbers on the number line. Points are plotted with error bars.

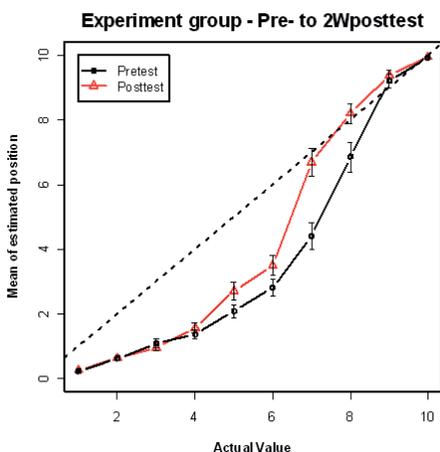


Figure 6. The figure shows the average result of the experiment group positioning the numbers on the number line. Points are plotted with error bars.

The fact that the results are neither linear nor logarithmic is surprising. Inspections of individual subjects (see Figure 8 later in the paper) show that individual estimates have strong linear trends, only not with the right slopes. This suggests that subjects use some sort of counting strategy, but with a counting unit that is not a tenth of the whole line, but rather a smaller unit.

Figures 5 and 6 suggest that the experimental manipulation was indeed successful. To analyze this we performed a two-way Anova with the summed error as the dependent variable and condition and pre- vs. posttest as independent variables. This produces an interaction effect between condition and test, $F(1,1441)=6.02$, $p=0.014$, and a main effect of test, $F(1,1441)=7.84$, $p=0.005$, but no main effect of condition, $F<1$. This means that the experimental group does indeed improve more on the posttest than the control group. Figure 6 shows that this improvement is mainly on the numbers 5 through 8.

To have a better picture of individual differences in the learning process, we used the k-means clustering algorithm (MacQueen, 1967) with as input the difference between the pre- and post-test of the accuracies of each of the ten numbers. The result of the cluster analysis of this combined group of both experiment group and control group indicates that there are two different patterns of improvement: one for numbers around five and six, and one for the numbers around eight (Figure 7).

Individuals in the first cluster (red circles) include six subjects from the experiment group and only two from the control group. This indicates that several more individuals in the experiment group have made improvement on the numbers 5-7 than those in the control group. The second cluster (green triangles) corresponds to no or little improvement, and includes 13 subjects from the control group and 8 subjects from the experiment group. And in the third cluster (black plus signs) there are three subjects from the control group and five from the experiment group.

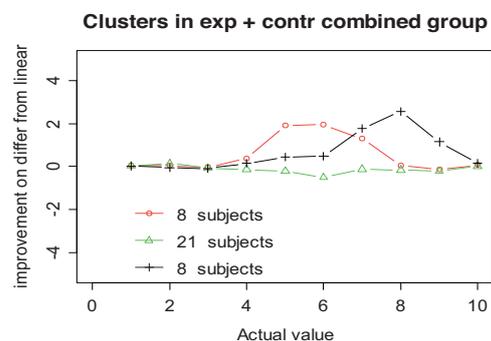


Figure 7. The graph shows the result of the cluster analysis on the improvement of distance from a true linear representation between pretest and posttest. Positive values indicate improvement and negative values the opposite.

To conclude, the data tell us a number of things. First, some sort of counting seems to be used to arrive at a point, but not with the correct counting unit. Second, improvements in performance seem to be centered around the middle point of the number line and towards the end of the number line, but hardly at the beginning. This suggests that subjects do not improve the length of their counting unit, but rather in the way they use it. Improvements around the middle of the number line suggest they learn that five is in the middle of the line and can be used as a starting point for counting. Improvements towards the end of the line suggests subjects learn that the higher numbers, 7 and 8 in particular, are close to 10, so that counting back from 10 is a better strategy than counting up from zero.

The model

A possible model of progressing towards a linear time scale can therefore be one that increasingly learns the locations of particular points on the number line, and uses those as anchors to determine the points that it does not know. It therefore needs some sort of representation of the positions of anchor points, but also a method for determining points in between those anchors by counting.

As a theory of how anchor points are stored in memory, we use ACT-R's declarative memory (Anderson, 2007). In order to determine positions between the anchor points, we use two mechanisms. The first one is a retrieve function that decides which anchor point will be the starting point. The second one is a count mechanism that uses a count unit to count up or down from the starting point to decide the position for the number on the number line. The initial size of the count unit is decided by average size from real data and randomly varies in size according to variation found between subjects in real data.

The details of the model

The basic assumption of the model is that the subjects already know how to count from 1 to 10, but that they have incomplete knowledge of how to put those numbers on the number line. An anchor point represents knowledge about putting numbers on the line and is expressed by associations between a number and a position on the line. In most cases this knowledge only consists of the number zero and the number ten on the extreme ends of the line, but may also consist of the middle point five.

To represent the different levels of knowledge about numbers and anchor points, we vary the base-level activation of the chunks associated with them.

If the model has to put a particular number on the number line, it tries to retrieve an anchor point from declarative memory for the number. If there is no direct match between any of the available anchors, to process of partial matching will retrieve the anchor point that has the highest activation. This activation depends on two aspects: the base-level activation of that anchor, and its similarity to the request number. So if the model tries to retrieve the number 6 and only 0 and 10 are available as anchors, the

model might retrieve 10 because it is closer to 6, but also 0 because that point has a higher base-level activation.

Whenever the model retrieves an anchor that is not already the number that it is trying to retrieve, it will apply counting to reach the desired point on the line. However, the unit of counting, following our data, is smaller than an actual tenth of the length of the line ($0.42 \text{ cm} \pm 20\%$). By simply varying the base-level activations of the anchors, we can reproduce most of the patterns of responses that we see in the data.

The model has an activation baseline function and there are three functions dealing with the declarative memory.

One function makes a reference list of numbers involved and their position on the number line. A chunk is represented as a list, with a number (what number is it about) and a position (where is it on the number line), and a reference list with moments in time the chunk has been accessed.

The mismatch function is based on Weber's law, and the result value is zero, a negative value or a positive value depending on whether the first number is similar, smaller than or larger than the second number. The mismatch assumes two numbers are more similar if they are closer and higher and is used to calculate the activation of a chunk.

A retrieval function is performing the retrieval and adds noise. Because of that we do not use the regular ACT-R retrieval rule and noise activation function.

Another function takes care of the counting procedure. The counting unit has an initial length correlating to the mean of the length of count units found in the real life data set from our experiment. The length of this unit is randomly shorter or longer for each individual simulation according to variation found in real life data. The same is done with the count unit for each count step.

In this same function simple proportioning is implemented the way that proportioning is activated after simulation of an individual's trial number 150.

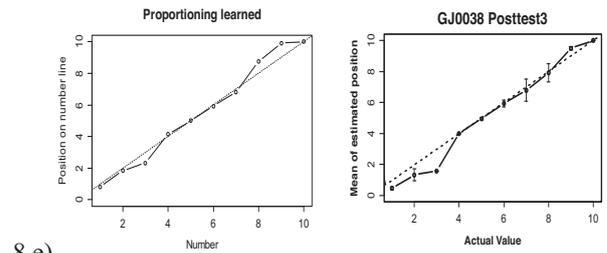
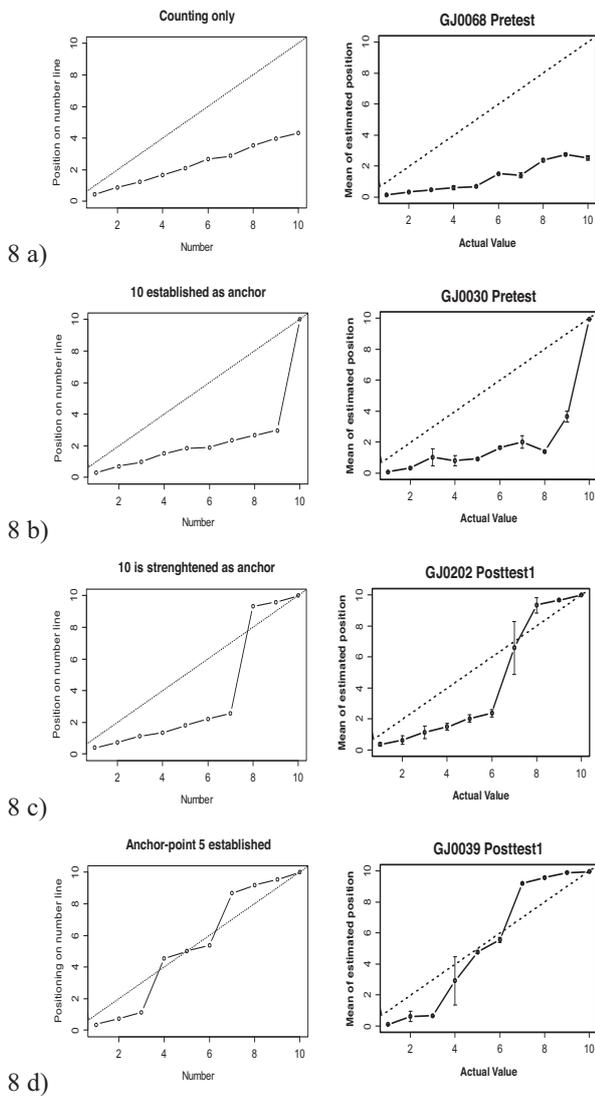
Results from running the model and discussion

When we run the model simulating a subject doing a certain number of trials, basic-level activation is not increased after every trial. The trials represent the training with the board game in real life. We assume that only after dozens of times of playing this game a subject obtains the kind of new crucial knowledge that makes a shift in numerical representation on the number line. This new knowledge could be that the position of a certain number is either at the beginning or at the end of the number line. In our experiment those numbers are 1 (or 0) and 10, the start point and the endpoint. So for the number at the endpoint there is no need for counting upwards from 0 anymore. We have got what we call a representational shift and the number at the endpoint has got a stronger activation as anchor point. That is why we let the model also run dozens of trials before each increase of base-level activation. The increase of base-level activation is done by

adding entries in the chunk for the appropriate anchor point.

Another example of such a representational shift is when a subject realizes that one or several numbers are close to 10 and counting downwards from 10 is how to position those numbers on the number line.

To what extent those shifts in knowledge represent different levels of knowledge is not clear, but we have made the assumption from our rather limited amount of real data that it could be that the first shift for children that have already learned to count, is to learn that the endpoint of the line is useful as an anchor point. In our experiment that is 10. Next is that some numbers are close to ten, then that five is an anchor point, and last, that the counting unit has to be adapted to a reasonable size. Plotted images show the relation and progress between these shifts. And we can easily find related and rather similar images to each of those steps from values of individuals from empirical data, when plotted (See figure 8).



8 e)

Figure 8 Model result from typical levels shown by model to the left and corresponding example from real data to the right: a): 1(or 0) is the only anchor point. b): Now knowing 10 as anchor point, c): Now knowing 10 even better, d): also knowing 5 as anchor point. e): Proportioning is activated.

When it comes to at what point a shift in knowledge should occur, in the model we have defined an amount of trials that we from our real data think is reasonably close to what we could find in real life.

A prior level of knowing how to represent numbers on an empty number line is of course when a child does not know how to do it at all. Siegler and Ramani (2008) show that even those at this prior level learned to deal with the number line during training with the board game. But for our model we have defined as the initial level when children know where 1 (or 0) is at the number line, and use counting only as strategy for putting other numbers on the right position.

The initial level, shown in figure 8 a), is a level where only counting is involved and base-level activation only on the chunk for the number 1 as an anchor point with a value of 1,15. In this case the chunk for 10 only has a base-level activation of -0,458.

At the next stage, shown in figure 8 b), which is after 60 trials, the base-level activation for the chunk of the number 1 is unchanged but for 10 it is increased to 0,640. The model now simulates where the anchor point 10 is, just like the subject GJ0030 at Pretest now knows where it is. .

After 100 trials we assume that a new shift occurs, shown in figure 8 c), Now the base level-activation for 1 is increased to 1,333 and for 10 to 0,928. Just like the subject GJ0202 in Posttest1 now knows, the model now simulates that 8 and 9 is close to 10 and positions those numbers by counting down from 10 as anchor point. The next shift will occur in the model after 150 trials, shown in figure 8 d). Base-level activation for 1 is unchanged, for 10 it is increased to 1,151. The number 5 now, as a new anchor point, has a base-level activation of 1,151. And the model now simulates knowing the midpoint, which is five, as anchor point. In real life data we find a close case in subject GJ0039 at Posttest1.

The last shift implemented in our model so far, shown in figure 8 e), is when a subject obtains knowledge about the need for, and how to, adapt the counting unit to the most suitable size, In this case the base level activation is

unchanged for all three anchor points, but proportioning of the counting unit is activated with some random errors.

The proportioning function of the model is rather preliminary and simply divides the physical length of the number line with the amount of numbers on line, which is 10 for this actual experiment, and adjusts it for error by randomizing according to what we find in real life data. Young children, as those in our experiment, do normally not obtain this level, and our experiment does not give us data for this. But we assume that what happens in real life is that finding a close to perfect size of the counting unit, is obtained either during training by trial and error or by dividing the line length in halves or thirds, one or several times.

In our data it seems that all of the subjects who understand the task use counting as an important part of the strategies for estimation. As we can see, in the same way as the results of our collected data from 6 year old children showed, we obtain no logarithmic curve from running our model. If we investigate the physical size of the unit used by the subjects in counting up or down from an anchor point, it is for all of them much smaller than a tenth of 25 cm, which was the length of the number line used in the estimation task. But on the opposite, with a larger scale, for example up to 100, the child's unit will be too large, and counting will often lead to a logarithmic curve like Siegler and others has found.

This shows that for the counting strategy, most of the subjects do not have a clear clue of what the size of a unit should be.

A last comment to our model, is that obviously there are moments between those representational shifts that we have built our model on so far, where subjects in real life obtain brick stones of knowledge that prepare for the shifts. For example we assume that when playing the board game and moving from number to number, the subjects learn connections between numbers. And the activation of those connections may be strengthened almost every time they play the game. This issue is in focus for further development of the model.

Conclusion

From our findings in real data we have concluded that a logarithmic scale for a representation of the result of a number line task depends on the proportion between the counting units the individual uses and the length of the empty number line. In our experiment the unit is too small to lead to a logarithmic representation.

We found that what they actually learn from training with the board game, are that higher numbers are close to 10 and that 5 and 6 are approximately in the middle of the line.

It does not make sense to show average data from the real life data set, because the individuals are so different. However, we can find shifts in learning levels in different individuals.

Those different shifts are easily simulated by our model.

Acknowledgments

Thanks to Gjesdal municipality for permission to conduct, and funding of our experiment, provided by paying teachers (for putting in extra hours) to assist in our experiment.

References

- Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* (p. 290). Oxford University Press. Retrieved February 1, 2011
- Barth, H. C., & Paladino, A. M. (2011). The development of numerical estimation: evidence against a representational shift. *Developmental science*, 14(1), 125-35
- Lende, L. K., & Taatgen, N. A. (2011). Modeling the Logarithmic-To-Linear Shift In Representations Of Numerical Magnitudes. In L. Carlson, C. Hoelscher & T. F. Shipley (Eds.), *Proceedings of the 33rd Annual Conference of the Cognitive Science Society* (pp. 207-212). Austin: TX: Cognitive Science Society.
- MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In L. M. Le Cam & J. Neyman (Eds.), *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability* (pp. 281-297). Berkeley, CA: University of California Press.
- Schneider, M., Heine, a, Thaler, V., Torbeyns, J., Desmedt, B., Verschaffel, L., et al. (2008). A validation of eye movements as a measure of elementary school children's developing number sense. *Cognitive Development*, 23(3), 409-422.
- Siegler, R. S., & Ramani, G. B. (2008). Playing linear numerical board games promotes low-income children's numerical development. *Developmental science*, 11(5), 655-61.
- Siegler, R. S., Thompson, C. a, & Opfer, J. E. (2009). The Logarithmic-To-Linear Shift: One Learning Sequence, Many Tasks, Many Time Scales. *Mind, Brain, and Education*, 3(3), 143-150.

Oscillatory Basis of Individual Differences in Working Memory Capacity

Krzysztof Andrelczyk (kandrelczyk@gmail.com)

Institute of Psychology, Jagiellonian University
Mickiewicza 3, 31-120 Krakow, Poland

Tomasz Smolen (tsmolen@up.krakow.pl)

Pedagogical University
Podchorazych 2, 30-084 Krakow, Poland

Adam Chuderski (adam.chuderski@uj.edu.pl)

Institute of Philosophy, Jagiellonian University
Grodzka 52, 31-044 Krakow, Poland

Abstract

The paper presents a novel formal model of the active buffer of working memory. The model uses synchronic oscillations in order to bind an item and its corresponding context into one representation, while asynchronic oscillations allow the model to maintain several separate representations. Due to bindings, the model exerts proper control over the buffer's contents, as demonstrated by effective rejection of distractors. Most importantly, the model predicts an inherent limitation in WM capacity that arises from the trade-off between the number versus the stability of representations bound by oscillations, which depends on the strength of lateral inhibition present among oscillating items. The systematic variation in inhibition leads to exact replication of the capacity distribution observed in a large sample of participants, as well as to prediction of a few novel, capacity-related experimental effects.

Introduction

Working memory (WM) is a neurocognitive mechanism responsible for the active maintenance of information for the purpose of its ongoing processing. It plays a crucial role in many complex cognitive processes like relational reasoning, problem solving, language, and learning (Jarrold & Towse, 2006). One of the most important features of WM is its heavily limited capacity. Usually, a person can maintain two up to six items in WM, with a mean individual capacity equaling four items (Cowan, 2001).

Some theories (Cowan, 2001; Oberauer, Süß, Wilhelm, & Sander, 2007) predict that WM consists of two distinct structures: a highly active and accessible buffer called *the focus of attention* (or *primary memory*, PM), and a less accessible *activated long-term memory* (or *secondary memory*, SM). It is argued that only PM is capacity limited, while SM is not, and that this very limit influences human performance on various tasks.

The most promising theoretical approach to storage in PM explains it as some kind of a pattern of oscillations. Several oscillatory models describe in a neurally plausible way the PM mechanisms which use patterns of fast, repetitive changes in activity of stored representations (i.e., use more than one oscillation during retention time) for coding items (Edin et al., 2009; Horn & Usher, 1992; Jensen & Lisman,

1998; Usher, Cohen, Haarmann, & Horn, 2001) and binding together different features of a maintained item (Hummel & Holyoak, 2003; Raffone & Wolters, 1998). Such models generated numerous predictions supported by neuroimaging data (e.g., Edin et al., 2009; Jensen & Lisman, 1998; Raffone & Wolters, 1998). They also showed that temporary bindings are crucial for complex cognition because they allow for representing arbitrary relational structures (Hummel & Holyoak, 2003; Oberauer et al., 2007).

Finally, oscillatory models (e.g., Hummel & Holyoak, 2003; Jensen & Lisman, 1998; Raffone & Wolters, 1998; Usher et al., 2001) naturally explain capacity limits as an emergent property of PM, which results from the trade-off between the number of to-be-maintained representations versus the ability to distinguish among them. As brain uses temporal coding for separating representations in PM, and time is a very limited resource, brain is not able to pack too many oscillations into one interval, because they start to overlap and so they stop being distinctive (=informative).

On the contrary, in models which do not rely on oscillations, one has to set a limit on the number of PM's slots (e.g., Kahana, 1996) or the amount of PM's activation (e.g., Daily, Lovett, & Reder, 2001) in an arbitrary way, so no natural capacity limit is being explained. Similarly, models which use only one cycle of activation change to code an item (e.g., Botvinick & Plaut, 2006; Davelaar, Goshen-Gottstein, Ashkenazi, Haarmann, & Usher, 2005), seem to be less neurally and functionally plausible than the full-fledged oscillatory models.

Although the oscillatory models nicely explain how brains handle maintenance in PM, and they give important insights into the nature of capacity limits, no such model has yet dealt with the fact that *people differ in capacity*. Though in principle we all could have had the maximum possible capacity, in fact capacity is hugely varied among humans.

This paper presents a novel formal model of PM. It aims to demonstrate which features (i.e., parameters) of the model's oscillatory mechanism are responsible for the observed individual differences in PM capacity. We test if a systematic manipulation to one such parameter, namely the strength of lateral inhibition applied among memory items, can replicate the distribution of capacity estimates in human population as well as a number of more specific effects.

Oscillatory model of primary memory

The main part of the model is a buffer, which contains a certain number of elements. Each element roughly approximates a neuronal assembly representing one specific feature of the world (e.g., an object's attribute, a concept, a word). As in many other models, a level of internal activation x_i , which falls in $[0, 1]$ range, is assigned to each element i .

The external output y of the element i in time t has been defined using a commonly applied sigmoid function of x_i , according to the following formula (1):

$$y_i[t] = \frac{1}{1 + \exp(-\delta(x_i[t] - .5))}$$

Parameter δ controls the level of nonlinearity of the relation between y and x . For small δ values this relation for $0 < x < 1$ is almost linear. With increasing δ , (1) gradually alters into a threshold function with the threshold at $x = .5$.

In order to express the presumed mechanism responsible for binding the features of one item while keeping the different items disjoint, we introduced a completely new (i.e., in comparison to other existing oscillatory models) equation, which controls changes in levels of activation (2):

$$x_i[t+1] = x_i[t] + \frac{\lambda}{1 + y_i[t]} + \alpha \sum_k \exp(x_k[t] - x_i[t]) - \beta \sum_j \exp(x_j[t] - x_i[t]) + \varepsilon(n)$$

Parameter λ controls how much element i is autoactivated by the recurrent connections feeding its output back into it, what reflects a commonly postulated self-recurrent nature of neuronal groups in brain structures underlying the focus of attention (e.g., O'Reilly & Munakata, 2000). Parameter λ primarily regulates the frequency of oscillations.

Index k denotes elements which output just before element i does, namely those in $[y_i, y_i + \kappa]$ range. So, parameter α determines how much the outputs of elements, which oscillate close to element i , increase its activation. This accounts for the known fact that neurons which fire in synchrony with a given neuron strongly influence its potential. Such a mechanism of coactivation helps to maintain synchrony among items with similar outputs. Parameter κ defines also the temporal resolution of bindings: the larger κ , the more distant (in terms of activation) elements will be considered by the model as bound within the same representation, namely those in $[y_i - \kappa, y_i + \kappa]$ range.

Index j denotes elements which are not k nor i elements, namely those that fall out of the above range. These elements encode representations separate from a representation encoded by the elements i and k . Parameter β controls the strength of inhibition exerted by elements j , which decreases the activation of element i . How much element j inhibits element i depends on a difference in the elements' activity: a relatively more active element will inhibit element i more strongly than will do a less active one. The last part of equation (2) consists of a noise ε , which is being drawn

from the normal distribution with the mean equaling zero, and the variance dependent on parameter n .

The activations and external outputs of elements are updated in discrete cycles. Each cycle represents a period of several milliseconds, though precise timings of the model's operation were not reported in this paper. As soon as output of an element reaches unity (this reflects firing of a neuronal group), the parameter λ for that element is temporarily changed to a relatively large negative value, which makes this element quickly fall below a base level of activation (set in the model to .2). This is meant to reflect the phenomenon of refraction. Then, the value of λ is being reset to a default value and the element starts building up its activation above the base level. However, inhibition signals may be so strong that the activation may decrease below zero – a value adopted as a minimal activation necessary to stay in the buffer – and the element permanently falls out of it.

Generally, the number of elements that can be bound together within one synchronic oscillation is not limited. However, in the following simulations we apply only pairs of synchronized elements (an item identity and its position).

Workings of the oscillatory model

The aim of the model is to maintain as many separate oscillations as necessary, for a given interval. Two elements making one oscillating pair (e.g., a letter and its temporal or spatial position, see below) are added to the buffer in the same time. The first pair is added with a random level of activation. Subsequent pairs can be added when activations of all other pairs $\forall x < 1 - 4 \times \kappa$. They are being added at a level of $x = x_{max} + \kappa + (1 - x_{max}) / 2$, where x_{max} denotes x value of the most active pair. This mechanism checks if there is enough place in activation space for new elements, and grants that at least on entering the buffer new pairs will be sufficiently distinctive from all other pairs.

In the model, the capacity limit arises because addition of consecutive pairs increases the strength of total inhibition that each pair receives. When this value surpasses the results of autoactivation (regulated by parameter λ) and coactivation (governed by parameter α), the elements with the lowest activation levels start falling out of the buffer. If one element from the pair falls out, then the coactivation is no longer possible, and the chance that the other element from that pair would also fall out drastically increases. Thus, the parameter β is the main determinant of the model's capacity. The higher β , the faster the elements start falling out of the buffer. So, the model predicts that a maximum capacity will be achieved when there is no inhibition at all ($\beta = 0$). Indeed, in such a case, the model was able to maintain twelve pairs, surpassing human capacity, but only when the noise was switched off. In more realistic cases, a certain amount of inhibition is necessary because it secures that oscillations will evenly occupy a respective time interval, helping to separate them. So, the most appropriate values of β reflect the trade-off between low (many elements can be maintained, but they are unbound) and high (less elements can be maintained, but they are properly bound) inhibition.

By gradually increasing the moderate value of β , we replicated the highest capacity (around five items) observed among people (see the next section), mean capacity (around

three items), and – the lowest possible capacity (one item). Respective patterns of oscillations are presented in Fig. 1.

Although β is the most important determinant of the model's capacity, we note that three other parameters can in principle modulate workings of the model. Firstly, the increase in parameter α would strengthen synchrony of bound elements. Such a mechanism may reflect a top-down boosting applied by the prefrontal cortex, which can pass additional activation to PM (see Edin et al., 2009). This extra boosting makes all elements more strongly activate each other. However, because we assume that a given element, when fires, can activate only elements firing in its temporal proximity, we expect that the boosting influences only the mechanism of coactivation. Another factor which impacts capacity is the level of noise (n). The higher noise, the higher is the probability that pairs get desynchronized. The noise may reflect numerous distinct factors, as fatigue, mental retardation, influence of drugs, etc. The last parameter related to capacity is the value of κ . If κ is large, then the pairs are stable, but there is little "room" for adding new pairs and the capacity is low. If κ is very low, in theory many pairs could be added, but even tiny differences in paired elements' activity make the pair desynchronize.

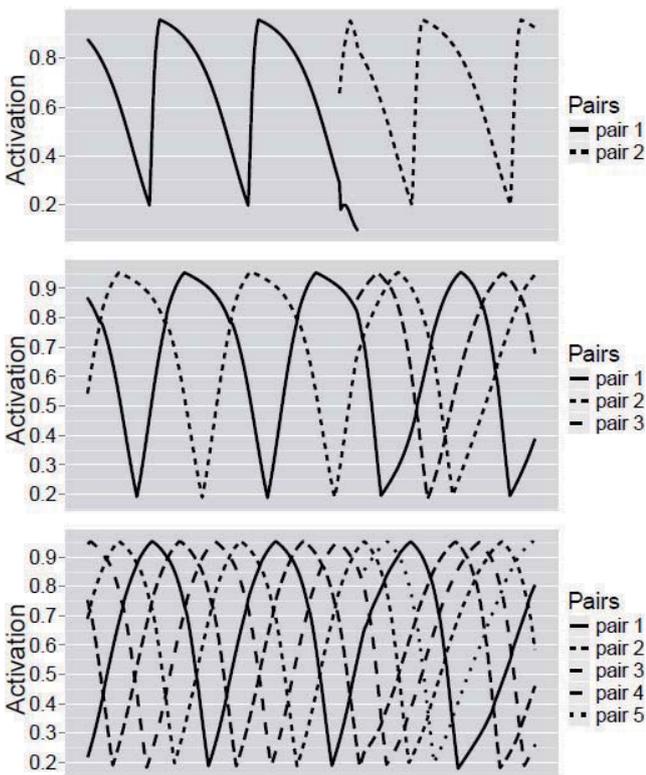


Fig.1: Patterns of oscillations for the lowest (=one item; upper panel), medium (=three items; middle panel), and largest (=five items; bottom panel) capacity. When capacity was insufficient, addition of a new pair eliminated an existing pair (see upper panel).

Measurement and simulation of individual differences in primary memory capacity

In the following simulations, we set α parameter to a low arbitrary value of .0001. Regarding κ , changing its values between .03 and .07 did not influence the model's capacity, so we set the κ value to .05. Parameter n was set to zero (i.e., noise was turned off). Parameter λ was drawn from a normal distribution which optimized the model's capacity given the adopted range of parameter β . So, in total we adjusted four global parameters. In order to replicate the distribution of WM capacity as observed in a sample of participants, we individually varied the values of β (see below).

We modeled two similar WM tasks. In the first one (the Sternberg task), the model attempted to add to its buffer several letter-position pairs, and then a probe in a particular position was presented. The model ran two processes which tested (a) whether the element identical to the probe could be found in the buffer and if a position bound to it (if any) matched the position of the probe or not, and (b) whether the element identical to the probe's position could be found and if an element bound to it matched the probe's identity or not. If either the identity or the position was found, and its binding matched either the probe's position or identity, respectively, then the model generated a positive answer. If both elements were found, but one of them did not match the corresponding element, the answer was negative. If any of two elements were not found, the model guessed either the positive answer with probability ρ (a decisional bias) or a negative answer with a chance $1 - \rho$.

Basic effects regarding primary memory

We started testing our model by checking if it is able to replicate the recency effect. When WM performance relies primarily on PM, as for example in the Sternberg task applied with fast presentation rate, an increased accuracy is observed for the most recent items in comparison to middle items, but there is no primacy effect (Chuderski, Stettner, Orzechowski, 2007). The simulated effect is shown in Fig.2.

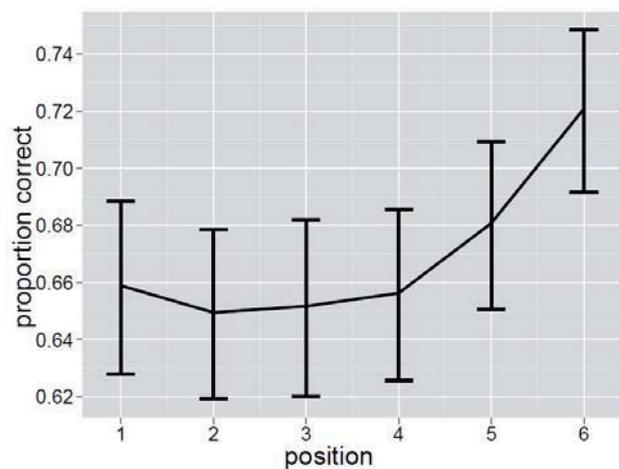


Figure 2: Accuracy of the model in all serial conditions of the six-item Sternberg task.

A more interesting observation – in light of the aim of this paper – regards the fact that people quite effectively use positional information to reject distractors presented to them. For example, one of us (Chuderski & Stettner, in revision, Exp. 1, positive digit condition) used a modified Sternberg task, which was analogical to the standard version with one exception that a probe (a letter) was accompanied by a digit, which denoted the letter’s position in a memory set. The digit could match the target’s position or not, and the task of 47 participants and 47 corresponding simulations was to accept only matching digits. We observed that participants correctly accepted more matching digits ($M = .78$, $SD = .11$) than incorrectly accepted non-matching ones ($M = .26$, $SD = .17$). This result indicates that they effectively maintained the positions of items in WM. Simulated results were close to observations: model accepted $M = .67$ of the matching digits, while it did not reject only $M = .33$ of the to-be-rejected digits. Slightly lower accuracy of the model resulted from the fact that it only used its PM, while people most probably relied their performance on both PM and SM. This result indicates that a proper PM model must account for binding of the representations with their contexts. An unbounded information may often be simply useless.

The distribution of primary memory capacity

The crucial simulation consisted of the replication of the distribution of PM capacity estimates, which had been observed in the sample of 168 young participants, who fulfilled a two-array comparison task (Luck & Vogel, 1997). The task is assumed to require maintenance of material in PM, while SM barely helps in doing this task due to the use of figural material and a fast presentation rate. Ten other participants were excluded from the original data because their results in the task suggested that they did not succeed to maintain even one item in their PM.

The original task required memorizing an array of a few items. Then, after a retention interval, the array was repeated, but there was 50% chance that one item was changed. The task was to indicate if the item had changed or not. We used a version of the task consisting on a single-probed recognition: one of the items in the second array was surrounded by a cue indicating that if any of the items had changed, it was only the surrounded one. The test included 90 trials. Each self-paced trial consisted of a virtual, four by four array filled on random with four, five, or six (i.e., set size) stimuli, being drawn from a pool of 16 simple black figures (e.g., a square, a circle, a rhombus, an arrow, a cross etc.), each approximately 2.5×2.5 cm in size. The array was presented for set size multiplied by 0.5 s.

An estimate of PM capacity uses the proportion of correct responses for arrays with one item changed (hits; H) and the proportion of incorrect responses for unchanged arrays (false alarms; FA). PM capacity is estimated to k items (out of N items of the set size), on the assumption that a participant produces a correct hit or avoids a false alarm only if a cued item is transferred to his or her PM (with the k/N chance). If a non-transferred item is cued, then a participant guesses the answer. Thus, the sheer PM capacity is equal to $k = N \times (H - FA)$. The value of k is believed to closely

approximate the actual number of items held in PM by an individual (Rouder, Morey, Morey, Cowan, 2011).

We used an analogous models as for the Sternberg task, with an exception that this time it encoded figures and their spatial positions. The value of $\beta = .0026$ allowed us to replicate the mean k value in the sample ($M k_{sim} = 3.01$, $M k_{obs} = 2.92$). In order to simulate 168 individual results we varied values of parameter β for each individual simulation, drawing it from the normal distribution with $M = .0026$ and $SD = .0004$. Histograms of the observed and simulated distributions of k values are presented in Fig. 3. Both distributions did not differ significantly ($\chi^2 = 6.97$, $df = 7$, $p = .431$). R^2 value for observed and simulated data was .93.

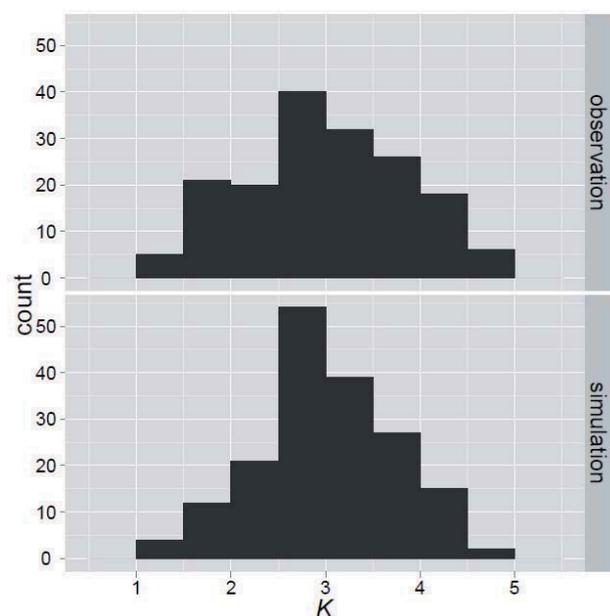


Figure 3: A number (count) of observations (upper panel) and simulations (lower panel) yielding particular k values.

Experimental effects related to capacity

Next, we examined if there were any specific differences in performing the task related to differences in participants’ capacity and – if yes – whether the model was able to predict them. Analysis of observed data indicated that participants more accurately responded to unchanged (congruent) arrays than to changed (incongruent) ones. In the model, this was accounted for by setting ρ value to .43. As parameter ρ regards guesses, and as highly capacious participants rarely guess (all necessary information is in their PM), the difference between accuracy in both conditions should diminish with increasing k value. This effect was found in both observed and simulated data (see Fig. 4).

Consequently, the model predicted that the discrepancy in accuracy between the incongruent and congruent conditions would be increasing as N increases (here, from four to six items). Such a pattern have also been found in observed data (see Fig. 5). However, it appeared that the model overpredicted accuracy in the four-items and five-items conditions, while it underpredicted accuracy in the six-item condition.

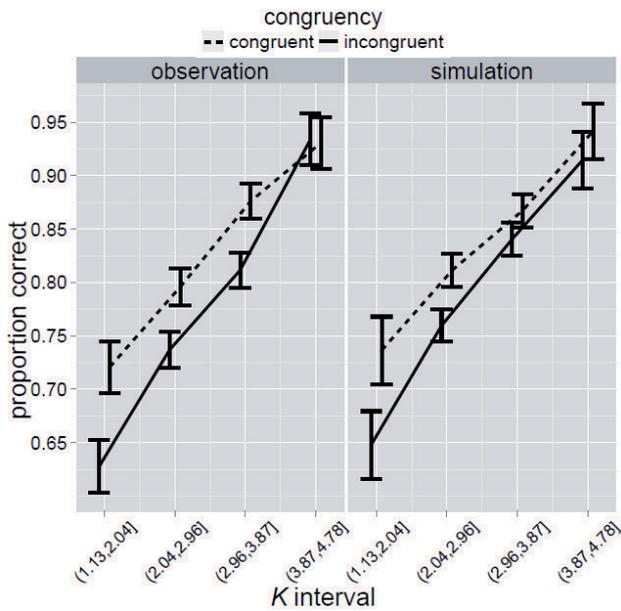


Figure 4: Accuracy in the congruent and incongruent conditions, in a function of the k value.

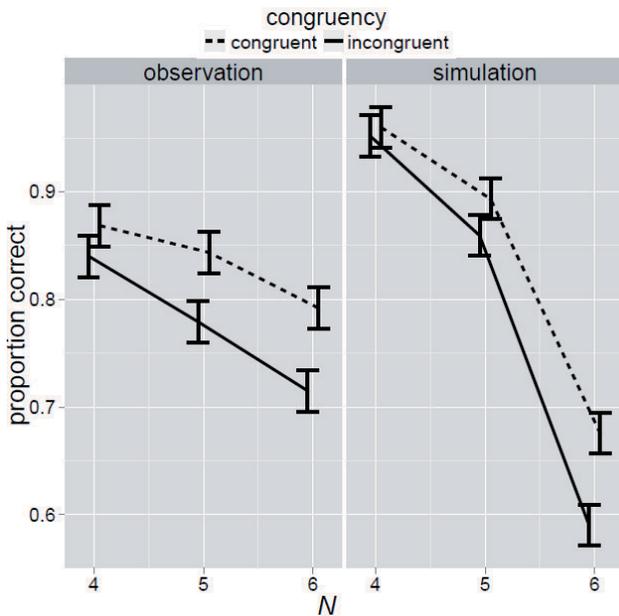


Figure 5: Accuracy in the congruent and incongruent conditions, in a function of the set size (N).

In search for a possible cause of the mismatch, we investigated if it can be related to the differences in capacity. We compared the k values for the respective item conditions between participants and between simulations yielding high versus low k estimates ($k > 3.5$ vs. $k < 2.5$). We found that in case of highly capacious participants, the k estimate significantly increased with N , $\Delta k = 0.85$, 95%CI = [0.58, 1.12], while in case of the low-capacity group the k value was significantly lower in the five-item condition than in the six-

item one, $\Delta k = -0.37$, 95%CI = [-0.09, -0.64] (see Fig. 6, left panel). The former effect is a direct consequence of the fact that k estimates of some participants equaled or surpassed four, so k value in the four-item condition underestimated their capacity. In fact, when only the six-items condition was considered, the maximum human capacity was $k = 5.6$, and was accounted for by the model. The latter effect is much more interesting: it indicates that PM of low capacity persons was even less effective than usual if the discrepancy between their actual capacity and the imposed requirements increased. This result is coherent with other behavioral and neuroimaging data (e.g., Todd & Marois, 2004).

The analysis of how the model coped with increasing N depending on adopted value of β determining its capacity, indicates that it showed qualitatively similar pattern of data (see Fig. 6, right panel, black lines), though there were substantial quantitative differences in comparison to observations. Increasing N value was on average not disruptive for the model's capacity in case of low values of lateral inhibition, why it dramatically decreased its capacity when the level of inhibition was high (i.e., when k was low).

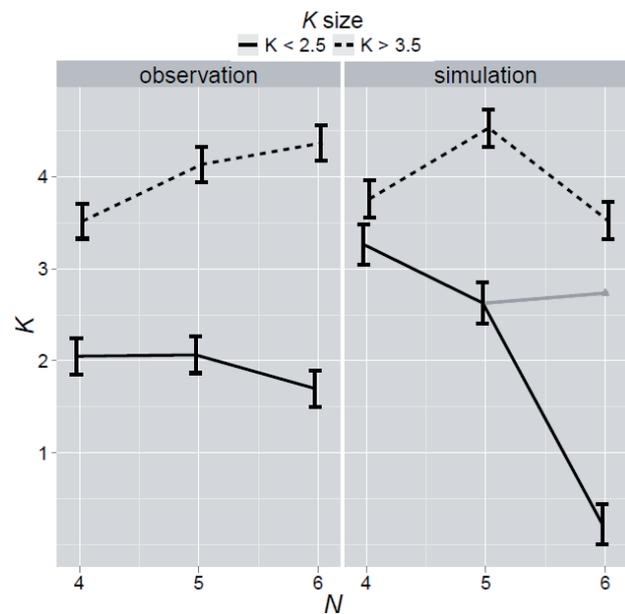


Figure 6: Mean values of k in a function of set size, for low-capacity (total $k < 2.5$) and high-capacity (total $k > 3.5$) participants/simulations.

What could be responsible for the model's more profound effect of the discrepancy between applied N value and individual k value? Edin et al. (2009) suggested that when such a discrepancy occurs, an additional top-down activation is recruited by the brain in order to counteract the lateral inhibition surpassing the brain's capability of dealing with it. We tested this hypothesis by re-running the six-item condition with twice as large α value (.0002) as in the original simulation. In result, the model's accuracy highly increased and our data better fitted human data (see Fig. 6, right panel, gray point). So, most probably our initial setting of α value underestimated the role of autoactivation in PM.

Discussion

Using a novel oscillatory model, which was aimed to reflect the mechanisms of active (attentional) maintenance of information in PM, the presented study has shown that variation in the strength of lateral inhibition among oscillating representations, which is necessary for formation and temporal separation of bindings among these representations, allowed for accounting for individual differences in PM capacity in large sample of participants. In our sample, it varied from around one item up to almost five items. It could even reach almost six items, when the largest set size was considered. This result seems to pose a serious problem to those oscillatory models, which predict that the maximum WM capacity is only four items (e.g., Raffone & Wolters, 1998). Most probably, these models would not be able to mimic the full distribution of WM capacity in human population. On the contrary, due to ability to coactivate the elements oscillating together, the presented model was able to maintain a dozen separate items at maximum, though when its parameters were being set on neurobiologically plausible levels (e.g., there was non-zero lateral inhibition), the model's capacity was naturally constrained to several items. In our view, the study suggests that the brain's ability to control (decrease) the level of inhibition within PM underpins its mechanisms supporting active maintenance of as much separate representations as possible.

However, the story regarding the replication of individual differences in capacity is not that simple. A more precise analysis of the effects of memory load in a function of individual capacity showed that the strength of lateral inhibition is not the only factor influencing the model's capacity. When the model attempted to maintain too much items in relation to its actual capacity, this increased the inhibition to such a high level that it led to a catastrophic decrease in capacity. In line with others (e.g., Edin et al., 2009; Todd & Marois, 2004), we suppose that the main role of the prefrontal cortex in active maintenance of information is to prevent such situations by additionally activating PM in a top-down manner. In our model, this was done by adjusting the coactivation of elements oscillating together. The analysis of computational properties of the coactivation and search for α values enhancing the fits of the model should be the subject of our future investigations.

Summing up, we presented a preliminary but highly original study on the neurocognitive mechanisms underlying the individual variation in PM capacity. Its results suggest that the concepts of oscillations and bindings can have a great explanatory power in regard to working memory.

Acknowledgments

This work was sponsored by The National Science Centre (NCN) of Poland (grant N106 417140).

References

- Botvinick, M. M., & Plaut, D. C. (2006). Short-term memory for serial order. *Psychological Review*, *113*, 201-233.
- Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, *24*, 87-114.
- Chuderski, A., & Stettner, Z. Access to working memory and its relation to reasoning. *Manuscript in revision*.
- Chuderski, A., Stettner, Z., & Orzechowski, J. (2007). Computational modeling of individual differences in short term memory search. *Cognitive Systems Research*, *8*, 161-173.
- Davelaar, E. J., Goshier-Gottstein, Y., Ashkenazi, A., Haarman, H. J., & Usher, M. (2005). The demise of short-term memory revisited: Empirical and computational investigations of recency effects. *Psychological Review*, *112*, 3-42.
- Daily, L. Z., Lovett, M. C., & Reder, M. L. (2001). Modeling individual differences in working memory performance: a source activation account. *Cognitive Science*, *25*, 315-353.
- Edin, F., Klingberg, T., Johansson, P., McNab, F., Tegner, J., & Compte, A. (2009). Mechanism for top-down control of working memory capacity. *PNAS*, *106*, 6802-7.
- Horn, D., & Usher, M. (1992). Oscillatory model of short term memory. In J. E. Moody, S. J., Hanson, & R. P. Lippmann (Eds.), *Advances in Neural Processing and Information Systems Vol. 4* (pp. 125-132). Morgan and Kaufmann.
- Hummel, J. E., & Holyoak, K. J. (2003). A symbolic-connectionist theory of relational inference and generalization. *Psychological Review*, *110*, 220-264.
- Jarrold, C., & Towse, J. N. (2006). Individual differences in working memory. *Neuroscience*, *139*, 39-50.
- Jensen, O., & Lisman, J. E., (1998). An oscillatory short-term memory buffer model can account for data on the Sternberg task. *Journal of Neuroscience*, *18*, 10688-99.
- Kahana, M. J. (1996). Associative retrieval processes in free recall. *Memory and Cognition*, *24*, 103-106.
- Luck, S. J., & Vogel, E. K. (1997). The capacity of visual working memory for features and conjunctions. *Nature*, *390*, 279-281.
- Oberauer, K., Süß, H.-M., Wilhelm, O., & Sander, N. (2007). Individual differences in working memory capacity and reasoning ability. In A. R. A. Conway, C. Jarrold, M. J. Kane, A. Miyake, & J. N. Towse (Eds.), *Variation in working memory* (pp. 49-75). Oxford: Oxford University Press.
- O'Reilly, R. C., & Munakata, Y. (2000). *Computational explorations in cognitive neuroscience*. Cambridge, MA: MIT Press.
- Raffone, A., & Wolters, G. (2001). A cortical mechanism for binding in visual memory. *Journal of Cognitive Neuroscience*, *13*, 766-785.
- Rouder, J. N., Morey, R. D., Morey, C. C., & Cowan, N. (2011). How to measure working memory capacity in the change detection paradigm. *Psychonomic Bulletin & Review*, *18*, 324-330.
- Todd, J. J., & Marois, R. (2004). Capacity limit of visual short-term memory in human posterior parietal cortex. *Nature*, *428*, 751-754.
- Usher, M., Cohen, J. D., Haarmann, H., & Horn D. (2001). Neural mechanism for the magical number 4: Competitive interactions and nonlinear oscillation. *Behavioral and Brain Sciences*, *24*, 151-152.

Modeling the Temporal Dynamics of Visual Working Memory

Johannes Lohmann (johannes.lohmann@uni-tuebingen.de)

Oliver Herbolt (oliver.herbolt@uni-tuebingen.de)

Martin V. Butz (butz@informatik.uni-tuebingen.de)

Cognitive Modeling, Sand 14
Tübingen, 72076 Germany

Abstract

Visual working memory (VWM) is one of the most crucial parts of the human cognitive system. Research focuses on the apparent limits in the capacity of this system and the reasons for them. But so far only a few formal models exist that can account for the temporal dynamics of the amount of information stored in VWM. We propose a combination of the well established theory of visual attention (TVA) with a dynamic memory model, resulting in an iterative, probabilistic framework for VWM. The model includes a consolidation as well as a decay mechanism and employs the strength concept to quantify the availability of a certain memory trace. We evaluate the model on available change detection data.

Keywords: VWM; TVA; Change Detection

Introduction

One of the main components of every cognitive task is the storage and maintenance of information in memory. Accordingly, research on working memory has a long tradition in both psychology and neuroscience (for an overview see Wixted, 2004b). Most models of working memory assume distinct systems for the preservation of verbal and visual information (Baddeley, 2003). Change detection tasks are frequently used to study the properties of the visual working memory (VWM). We focus on modeling VWM in this paper.

One way to investigate VWM is to ask participants to detect changes in subsequently presented displays. Early change detection studies (Phillips, 1974; Pashler, 1988) provided evidence that not only the amount but also the strength of information stored in VWM dynamically changes over time. For example, for short temporal delays between two subsequent displays change detection performance is very accurate, but it deteriorates for longer temporal delays. Pashler (1988) also found that change detection performance increased with longer presentation durations of the initial display. Apparently this is due to the fact that more information can be encoded the longer the initial display lasts. On the other hand it is possible that the encoded information becomes more stable for longer display durations. Seeing that VWM and visual perception additionally appear to be highly intertwined (Alvarez & Cavanagh, 2004; Gao, Gao, Li, Sun, & Shen, 2011), we approximate the dynamics of VWM in conjunction with visual perception by modeling encoding and memory consolidation with the same hypothetical process.

By now, only a few quantitative models are available that describe the process of stimulus encoding and the

preservation of the obtained information at once (but see Johnson, Spencer, & Schöner, 2009 for a neural model). Here, we provide a parsimonious quantitative model that can account for changes in the amount of stored information over time. To test the idea that memory encoding and VWM maintenance processes interact, we introduce a memory mechanism that also operates during the display presentation.

In the next section we give an outline of the model and sum up the underlying assumptions. Then we give a short description of TVA. After this we describe the features of our memory model in detail and give an example for the predictions of the complete model. We use the results of Phillips (1974) to evaluate our model. A short discussion concludes the paper.

General Assumptions

We investigate processing of visual stimuli at the stage of “perceptual units” (Bundesen, 1990). Perceptual units can be considered as segmented parts of the current visual input. Each unit can be described by feature dimensions like color or shape. The encoded categorizations of the feature dimensions are assumed to be the information that is stored in VWM. The theory of visual attention (TVA, Bundesen, 1990) captures this encoding stage in a formal framework.

We assume that every encoded categorization of a certain stimulus dimension can be described with a *strength* that quantifies the availability of the respective categorization. This strength is not constant but changes over time. Two processes affect the strength. First a consolidation process increases the strength over time. When the strength of a certain categorization increases during the presentation of a display, we refer to this process as on-line consolidation. Otherwise, we refer to consolidation as off-line consolidation. Seeing that consolidation generally takes place all the time, categorizations that are encoded earlier typically reach higher strength values than comparable types of categorizations that are encoded later on. Additionally, consolidation depends on memory load. Second a degradation process reduces the strength over time. We assume the degradation to take place at a constant rate after the offset of a certain stimulus display. Moreover, we assume this process to be independent of memory load. If the strength of a certain categorization falls below a threshold the respective categorization is removed from VWM. This is similar to

the idea of *some-or-none* representations proposed by Zhang and Luck (2009).

To sum up our model describes the following processes:

- Encoding of categorizations of stimulus dimensions (e.g. color or shape) from a display.
- On-Line consolidation of the stored categorizations during the presentation of the display.
- Off-Line consolidation of the stored categorizations after the offset of the display.
- Decay of stored categorizations after the offset of the display.

To test the resulting model the estimated memory load is transferred into a behavioral measure. We focus on modeling change detection performance. In the next sections we describe the different assumptions in more detail and provide the necessary formalizations.

Encoding of Information

Before we can consider the properties of information stored in VWM it is necessary to describe how information is encoded in the first place. TVA, proposed by Bundesen (1990), is a quantitative model of visual attention that accounts for a broad range of phenomena (Bundesen, Habekost, & Kyllingsbæk, 2005; Logan, 2002). TVA was successfully applied to model iconic memory (Sperling, 1960), visual search (Treisman & Gelade, 1980), switch costs (Logan, Schneider, & Bundesen, 2007), as well as attention deficits in clinical populations (Duncan et al., 1999). TVA allows quantitative predictions of the amount of information stored in VWM at a certain time. To model visual attention TVA integrates bottom-up processes (via the sensory properties of the relevant information) and top-down processes (via the intention to perform a task).

TVA models the encoding of visual stimuli in VWM as the combination of a *filtering* and a *pigeonholing* process. Filtering selects objects, whereas pigeonholing assigns categories to the selected objects. TVA proposes a race model where different categorizations compete for the incorporation in VWM. This race is formalized as the conditional probability of a categorization to be encoded, given that it was not encoded earlier. The following *rate equation* describes the probability that the categorization of item x belonging to category i enters VWM:

$$\nu(x, i) = \eta(x, i)\beta_i \frac{w_x}{\sum_{z \in T} w_z}, \quad (1)$$

where the categorization likelihood $\nu(x, i)$ depends on the sensory evidence $\eta(x, i)$ that object x belongs to category i , on the perceptual decision bias β_i for category i ,

and on the attentional weight w_x relative to all other attentional weight values w_z for all objects z in the visual display T .

The attentional weights w_z depend on the *pertinence* of a given categorization that can be considered as the subjective relevance of this categorization. For every object x in the visual field, an attentional weight is obtained by the following *weight equation*:

$$w_x = \sum_{j \in R} \eta(x, j)\pi_j, \quad (2)$$

where R denotes the set of all perceptual categories, $\eta(x, j)$ denotes the sensory evidence for element x belonging to category j , and π_j is a pertinence value for category j . The higher the pertinence of a certain category, the higher the likelihood to attend to objects that fall into the respective category.

TVA realizes filtering by attentional weights. If the task is to select red objects the value of π_{red} and the resulting weights for red objects would be high. Pigeonholing is biased by parameter β . If a task requires the categorization of letters rather than digits, the value of β would be higher for letters. The consequence of combined filtering and pigeonholing in this example would be the faster encoding of red letters compared to other stimuli.

Despite examples of successful applications, TVA cannot account for decay in the content of VWM. Bundesen (1990) assumed a fixed capacity VWM model that is filled by a Bernoulli process. As the capacity is fixed, it cannot account for any loss of stored information over time. Therefore we propose a dynamic model for the VWM, which also considers memory decay.

A dynamic Model of VWM

We assume a continuous mnemonic resource (Bays & Hussain, 2008; Wilken & Ma, 2004; Verghese, 2001) that is used to consolidate information in VWM¹. Memory limits emerge over time as it becomes more and more difficult to maintain the stored information. To describe the state of certain stored categorizations, we employ the *strength* concept proposed by Wickelgren (Wickelgren, 1974). According to this theory the availability of a certain memory trace can be described by its strength. This strength changes over time. Initially memory traces are weak but their strength increases over time. Hence older traces are more stable than younger ones (Jost's second law, see Wixted, 2004a). For short retention intervals even the weak traces can be preserved. For longer intervals only the strongest traces persist.²

¹Please note that the idea of fixed slots (Zhang & Luck, 2008) is a prominent alternative to this approach (see Fukuda, Awh, & Vogel, 2010 for a comparison of both approaches).

²Please note that Wickelgren assumed an ever decaying strength, an increase of strength that is realized by the con-

Initially the strength of an encoded categorization is set to 1.0. We assume three different processes that affect the strength of a certain stored categorization over time. First, a decay process applies to each stored categorization after stimulus display offset. Second, an on-line consolidation mechanism increases the strength of a categorization after its encoding throughout the presentation of a display. This assumption is based on Jost’s second law to give older traces the proposed advantage in durability. Third, off-line consolidation increases the strength after display offset. We now describe these processes in detail.

Decay

We assume the strength of every stored categorization to degrade at a constant pace after the offset of the stimulus display. The decrement is denoted as ξ , it is assumed to be a random variable and to be unaffected by the load of VWM. If a certain strength falls below 0, the according categorization is lost.

On-Line Consolidation

We assume the conditional encoding probability $\nu(x, i)$ to be proportional to the amount of mnemonic resources that are used to consolidate a certain categorization after it enters VWM. This assumption is justified as $\nu(x, i)$ reflects top-down influences, such as the relevance of a certain categorization (see Equation 1). Accordingly, we assume the strength $s(x, i)$ to increase by $\nu(x, i)$, each time consolidation applies. We furthermore assume that only one categorization is consolidated at a time, essentially assuming a serial consolidation process similar to the one discussed in Schneider (1999). The categorization that is consolidated next is chosen randomly (with replacement). Each consolidation may require several iterations.

Off-Line Consolidation

A similar mechanism like the proposed on-line consolidation is assumed to operate after the offset of the display. The difference between on-line and off-line consolidation is the value of the increment of a certain strength $s(x, i)$. We assume that the sum of all conditional encoding probabilities to be proportional to the amount of mnemonic resources, referred to as κ , that can be used for consolidation:

$$\kappa \propto \sum_{x \in T} \sum_{i \in R} \nu(x, i) \quad (3)$$

For off-line consolidation, we assume that this amount is distributed over the currently stored categorizations. If all categorizations are of equal relevance, the amount

solidation components of our model was never proposed by Wickelgren. Instead he assumed the decay to slow down over time.

of κ used to consolidate a single categorization can be obtained by dividing κ by the number of currently stored categorizations. In effect, the consolidation of a single categorization is more effective when less categorizations are stored in VWM. As a consequence, the assumed decay mechanism bears a stronger influence if VWM load is high.

Example

As an example, let’s assume a very simple stimulus display containing three objects with different colors. Let’s further assume that the colors of these objects are equally relevant for the current task, whereas other possible features are irrelevant. We now want to use our model to predict the temporal changes in VWM content. An exemplary complete time-course is displayed in Figure 1.

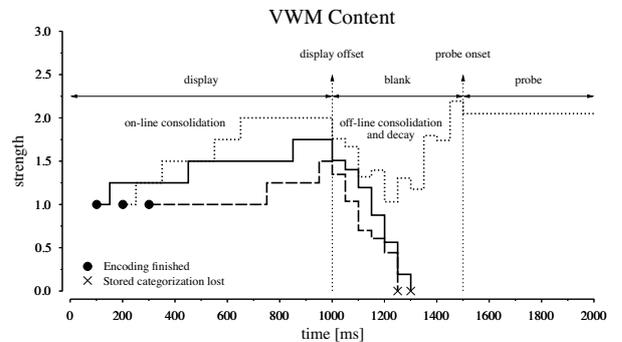


Figure 1: Example of the predicted changes in VWM over time.

Encoding. As only color information is relevant all the respective $\beta(x, color)$ and $\pi(x, color)$ would be high, whereas all other β and π values would be 0. At every time step each of the color categorizations is encoded with a certain probability ($\nu(x, color)$) and stored in VWM. As this is a probabilistic process, the time of successful encoding differs for the different categorizations (see Figure 1, the time of encoding is indicated by a filled circle).

On-Line Consolidation. As shown in Figure 1, the strength of the stored categorizations probabilistically increases over time during the presentation of the display. If only one categorization is stored, it is consolidated every time consolidation takes place. As only one categorization is consolidated at a time, the growth of an individual strength value declines as more and more categorizations are encoded.

Decay. After the offset of the display, the strengths of the stored categorizations decay. Here, we assume

a uniformly-distributed, noisy amount of decay per iteration. As exemplary shown in Figure 1, two categorizations are lost after about 300 ms due to the decay process.

Off-Line Consolidation. After the offset of the display, the decay in strength is encountered by an consolidation mechanism. The higher the memory load, the less effective is the consolidation mechanism. As more and more categorizations are lost, consolidation becomes more effective in protecting the remaining categorizations from further decay (see Figure 1, at around 1300 ms). In the current model, we do not consider an upper bound for memory strength.

Results

In this section we apply our model to the data from the first experiment reported by Phillips (1974). This study investigated change detection performance for lighted pixels in a square matrix. Each pixel in the matrix had a chance of 50% to be lit. The matrix was presented for one second. The matrix size varied between 4×4 , 6×6 and 8×8 pixels. After an inter stimulus interval (ISI) of either 20, 1000, 3000 or 9000 ms, a probe display appeared. In 50% of the trials, the probe display was equal to the initial display, in the other trials it differed with respect to one pixel. The participants had to indicate if the probe display differed from the initial one. Phillips (1974) collected the percentage of correct responses as a dependent measure. The results suggested two different types of storage systems. First, a high-capacity but short lasting iconic storage system. Second, a more persistent but capacity limited short term storage system. The data basis for the evaluation is quite small comprising only 12 mean detection probabilities. Nevertheless, the data pattern is challenging because of the broad variation in the length of ISIs, covering sensory memory as well as short term memory. This is the main reason for which we decide to use this data set instead of more contemporary studies like the one reported by Vogel, Woodman, and Luck (2001).

To obtain predictions of the percentage of correct responses, we transfer the estimated VWM load after the ISI into a probability. We set the predicted probability of success equal to the number of available traces for the relevant feature dimensions:

$$p_{detect} = \frac{E_{preserved}}{T} \quad (4)$$

where T refers to the number of relevant categorizations and $E_{preserved}$ denotes the number of stored categorizations in the relevant dimension. As the participants had to perform a same / different judgment, a correction for guessing is assumed. Usually the guessing probability should be 0.5:

$$p_{predicted} = p_{detect} + (1 - p_{detect}) \frac{1}{2} \quad (5)$$

This is similar to the formula proposed by Pashler (1988), except for the fact that we assume a constant guessing probability for all participants.

We used four different versions of our model to investigate if the different mechanisms improved the predictions. The first version modeled encoding with TVA and assumed constant decay during the ISI. Neither on-line consolidation nor off-line consolidation were used. The second version included the off-line consolidation process during the ISI. The third version included the on-line consolidation mechanism. The fourth version applied both off-line and on-line consolidation.

All η , β and π values were set to 1.0. Hence the ν values only differed between matrix sizes. The equality of the TVA parameters is plausible because the categorization of every pixel as either lit or not lit was equally relevant for the task. Due to our assumption of proportionality between κ and ν , only the decay parameter ξ was treated as a free parameter. We applied uniformly distributed noise to ξ , so that the actual decay rate could vary between 0 and ξ in each time step. Additionally, all models that applied on-line or off-line consolidation or both had an additional free parameter, referred to as *lag*, which determined how much iterations have to pass between two successive consolidation events. All free parameters were constant over matrix sizes. As encoding, consolidation, and decay were modeled as probabilistic processes, we averaged 50 independent runs for every iteration of the parameter estimation.

The predictions of the different versions are displayed in Figure 2. The results are the average of 50 independent runs, obtained with the best parameter sets with respect to RMSE.

Table 1: Parameter estimates, RMSEs and r^2 values for the different setups.

Setup	ξ	lag	RMSE	r^2
Decay only	0.020	-	0.154	.54
Off-Line consolidation	0.031	5	0.048	.92
On-Line consolidation	0.021	0	0.151	.58
Full model	0.030	5	0.046	.93

Results with respect to RMSE and declared variance (r^2) are displayed in Table 1. Additionally, the parameter estimates are included. The best results were obtained with models assuming decay and off-line consolidation. The addition of the on-line consolidation improved the fit only marginally. At least for this data pattern it seems not necessary to assume an on-line consolidation mechanism. This can also be concluded from

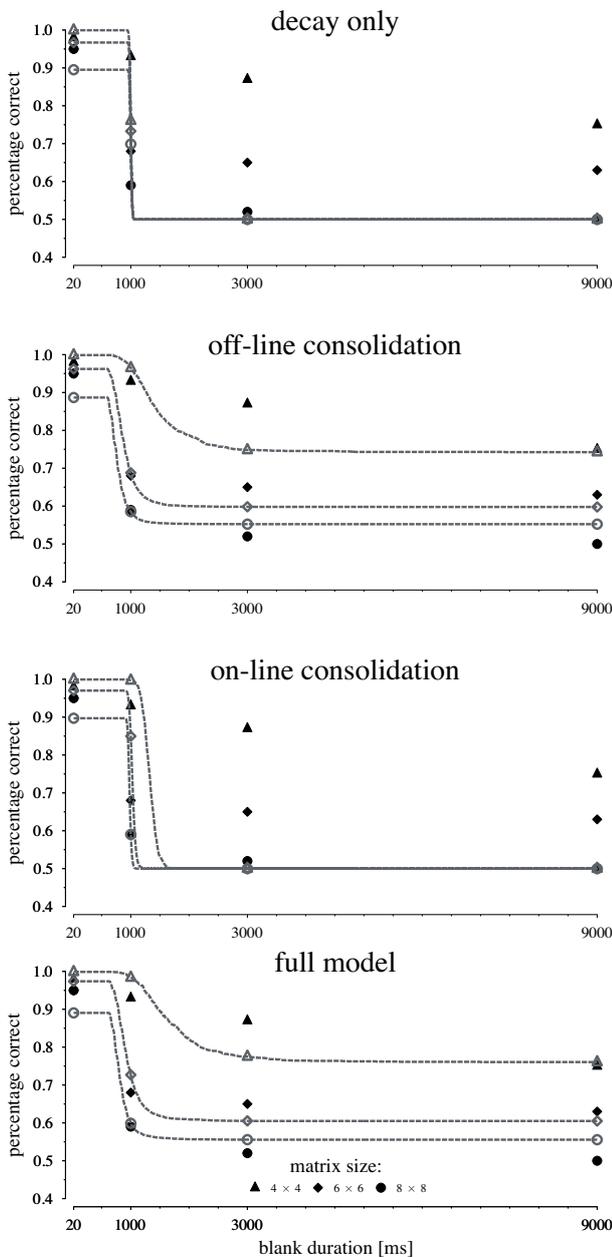


Figure 2: Results obtained with the different versions of our model. Different markers indicate the different matrix sizes. The predicted probabilities are indicated by dashed lines, the predictions for the measured ISIs are indicated by empty markers.

the parameter estimates. There are nearly no differences between the estimate of ξ between the model that only applied off-line consolidation and the full model, applying both on-line and off-line consolidation.

Discussion

We proposed a model that accounts for the temporal dynamics of the information stored in VWM. Therefore, we combined TVA with a dynamic memory model that assumes the concurrent operation of a consolidation and a decay mechanism. Our model combines the encoding mechanism proposed by TVA with the strength concept developed by Wickelgren. The assumed on-line consolidation was included to account for the age of different memory traces. It is in line with Josts second law (cf. Wixted, 2004a). However, the obtained results when modeling the data reported by Phillips (1974) indicate that this mechanism is not necessary. Only off-line consolidation was mandatory to achieve a reasonable data fit. Possibly it becomes more relevant for longer display durations. This would be plausible as Josts law was based on observations concerning long-term memory recall. Nonetheless, the data fit achieved only with the assumption that memory consolidation counteracts memory decay during display offset did yield a good data fit. Despite this success, certainly several aspects of the model ask for enhancements and further verifications.

First, our model is neutral about the source of memory degradation. The original strength theory proposed by Wickelgren assumed interference between successive stimuli to be the main source of memory degradation. The decay mechanism assumed in our model is more in line with trace decay, due to prolonged retention intervals. To account for interference, it would be necessary to account for the effect of successive stimulus onset. As it is displayed in Figure 1, currently the presentation of subsequent displays is not supposed to affect the VWM content. In the future, we will attempt to also account for findings that highlight the role of interference for VWM contents (Makovski, Sussman, & Jiang, 2008) with our model.

Second, the proposed on-line consolidation mechanism seems not to be necessary at least to account for the data reported by Phillips (1974). Possibly the display duration of 1000 ms applied by Phillips (1974) is too short to require the assumption of on-line consolidation. Therefore the model should be applied to change detection data that was obtained with longer display durations to further investigate the validity of the proposed on-line consolidation mechanism. Possible data for evaluation could be obtained from the experiments reported by Hollingworth and Henderson (2002).

Even with the mentioned shortcomings the results obtained with the model are promising. As the number of free parameters is very small, it is highly unlikely that this is due to the flexibility of the model. But especially the necessity of the assumed on-line consolidation mechanism remains unclear and has to be investigated in more detail in the future. Further extensions of the model could include a more detailed account for early visual

processing, for instance by a layered battery of gabor-filters as it was proposed by Serre, Wolf, Bileschi, Riesenhuber, and Poggio (2007). In this way, visual saliency effects may be modeled as well. Furthermore, the mechanisms by which the top-down control variables (β and π) are assignment could be modeled in more detail, possibly similar to the Bayesian feature- and location-based approach described in Chikkerur, Serre, Tan, and Poggio (2010). We expect that the further integration of such neural models will provide another bridge between cognitive psychology and neuroscience, combining the desirable features of both approaches: A detailed description of the involved processes and an output format that allows direct model evaluation based on observed data.

References

- Alvarez, G. A., & Cavanagh, P. (2004). The capacity of visual short-term memory is set both by visual information load and by number of objects. *Psychological Science, 15*, 106-111.
- Baddeley, A. (2003). Working memory: Looking back and looking forward. *Nature Reviews Neuroscience, 4*, 829-839.
- Bays, P. M., & Hussain, M. (2008). Dynamic shifts of limited working memory resources in human vision. *Science, 321*, 851-854.
- Bundesden, C. (1990). A theory of visual attention. *Psychological Review, 97*, 523-547.
- Bundesden, C., Habekost, T., & Kyllingsbæk, S. (2005). A neural theory of visual attention: Bridging cognition and neurophysiology. *Psychological Review, 112*, 291-328.
- Chikkerur, S., Serre, T., Tan, C., & Poggio, T. (2010). What and where: A bayesian inference theory of attention. *Vision Research, 55*, 2233-2247.
- Duncan, J., Bundesden, C., Olson, A., Humphreys, G., Chavda, S., & Shibuya, H. (1999). Systematic analysis of deficits in visual attention. *Journal of Experimental Psychology: General, 128*, 450-478.
- Fukuda, K., Awh, E., & Vogel, E. K. (2010). Discrete capacity limits in visual working memory. *Current Opinion in Neurobiology, 20*, 177-182.
- Gao, T., Gao, Z., Li, J., Sun, Z., & Shen, M. (2011). The perceptual root of object-based storage: An interactive model of perception and visual working memory. *Journal of Experimental Psychology: Human Perception and Performance, 37*, 1803-1823.
- Hollingworth, A., & Henderson, J. M. (2002). Accurate visual memory for previously attended objects in natural scenes. *Journal of Experimental Psychology: Human Perception and Performance, 28*, 113-136.
- Johnson, J. S., Spencer, J. P., & Schöner, G. (2009). A layered neural architecture for the consolidation, maintenance, and updating of representations in visual working memory. *Brain Research, 1299*, 17-32.
- Logan, G. D. (2002). An instance theory of attention and memory. *Psychological Review, 109*, 376-400.
- Logan, G. D., Schneider, D. W., & Bundesden, C. (2007). Still clever after all these years: Searching for the homunculus in explicitly cued task switching. *Journal of Experimental Psychology: Human Perception and Performance, 33*, 978-994.
- Makovski, T., Sussman, R., & Jiang, Y. V. (2008). Orienting attention in visual working memory reduces interference from memory probes. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 34*, 369-380.
- Pashler, H. (1988). Familiarity and visual change detection. *Perception & Psychophysics, 44*, 369-378.
- Phillips, W. A. (1974). On the distinction between sensory storage and short-term visual memory. *Perception & Psychophysics, 16*, 283-290.
- Schneider, W. X. (1999). Visual-spatial working memory, attention, and scene representation: A neurocognitive theory. *Psychological Research, 62*, 220-236.
- Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., & Poggio, T. (2007). Robust object recognition with cortex-like mechanisms. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*, 411-426.
- Sperling, G. (1960). The information available in brief visual presentations. *Psychological Monographs, 74(I 1, Whole No. 498)*, 1-29.
- Treisman, A., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology, 12*, 91-136.
- Verghese, P. (2001). Visual search and attention: A signal detection theory approach. *Neuron, 31*, 523-535.
- Vogel, E. K., Woodman, G. E., & Luck, S. J. (2001). Storage of features, conjunctions, and objects in visual working memory. *Journal of Experimental Psychology: Human Perception and Performance, 27*, 92-114.
- Wickelgren, W. A. (1974). Single-trace fragility theory of memory dynamics. *Memory & Cognition, 2*, 775-780.
- Wilken, P., & Ma, W. J. (2004). A detection theory account of change detection. *Journal of Vision, 4*, 1120-1135.
- Wixted, J. T. (2004a). On common ground: Jost's (1897) law of forgetting and Ribot's (1881) law of retrograde amnesia. *Psychological Review, 111*, 864-879.
- Wixted, J. T. (2004b). The psychology and neuroscience of forgetting. *Annual Review of Psychology, 55*, 235-269.
- Zhang, W., & Luck, S. J. (2008). Discrete fixed-resolution representations in visual working memory. *Nature, 453*, 233-235.
- Zhang, W., & Luck, S. J. (2009). Sudden death and gradual decay in visual working memory. *Psychological Science, 20*, 423-428.

A Reinforcement Learning Model of Bounded Optimal Strategy Learning

Xiuli Chen (xxc116@cs.bham.ac.uk)

School of Computer Science, The University of Birmingham
Edgbaston, Birmingham, B15 2TT, UK

Andrew Howes (A.Howes@cs.bham.ac.uk)

School of Computer Science, The University of Birmingham
Edgbaston, Birmingham, B15 2TT, UK

Abstract

In this paper we report a reinforcement learning model of how individuals learn the value of strategies for remembering. The model learns from experience about the changing speed and accuracy of memory strategies. The reward function was sensitive to the internal information processing constraints (limited working memory capacity) of the participants. In addition, because the value of strategies for remembering changed with practice, experience was discounted according to a recency-weighted function. The model was used to generate predictions of the behavioural data of 40 participants who were asked to copy appointment information from an email message to a calendar. The experience discounting parameter for a model of each individual participant was set so as to maximize the expected rewards for that participant. The predictions of this bounded optimal control model were compared with the observed data. The result suggests that people may be able to choose remembering strategies on the basis of optimally discounted past experience.

Keywords: bounded optimal; reinforcement learning; information processing bounds; memory constraints.

Introduction

Human beings are *bounded optimal* if they are able to maximize utility subject to the bounds imposed by their information processing capacities and their experience (Howes, Vera, Lewis and McCurdy 2004; Lewis, Vera and Howes, 2004; Howes, Lewis and Vera 2009). This paper reports progress towards a bounded optimal control theory of how people perform simple tasks that make use of memory. The model uses reinforcement learning to acquire optimal strategies given bounds imposed by short-term memory and experience. It therefore represents an example of a class of models that harness both the rigour of optimisation and theories of the bounds on human information processing (Anderson et al. 2004). The model also represents a departure from theories of unbounded optimisation (Griffiths & Tenenbaum 2006; Griffiths, Kemp and Tenenbaum 2008) and descriptive theories of bounds.

The model reported in the current paper captures what people choose to do given experience of the behavioural consequences of tasks that required memory. For example, when reading and writing a telephone number a person may choose to read the whole number, store it in memory, and then write it out. Alternatively he/she may choose to read the number 3 digits at a time and write out each 3-digit block before reading the next. There are many strategies but

each has potentially different performance characteristics: Some might be fast but generate many errors, others relatively slow but reliable. Tasks such as these have been investigated by Gray, Simms, Fu and Schoelles (2006). Gray et al. used the Blocks World task to study the choices that people make about what to remember. The participants were required to reproduce patterns of coloured blocks from a Target window to a Workspace window. For example, there might be 8 different coloured blocks which were positioned randomly in a 4x4 grid. The number of blocks encoded by a participant on each visit was regarded as corresponding to a strategy. Gray et al. demonstrated that participants were able to adapt their choices of strategy to the cost/benefit structure of the environment given experience.

More recently, Howes et al. (submitted) employed a similar task, called the Email-Calendar Copy task, in which the participants were required to copy the appointment information from an email interface to a calendar. The results suggested not only that participants were able to adapt their choice of strategy, as demonstrated by Gray et al. (2006) but also that many would end up preferring the optimal strategy given their learned knowledge. The reinforcement learning model reported in the current paper is a model of the results of Howes et al. (submitted). Unlike with many previous reinforcement learning models, including those of Gray et al. (2006), the current model parameters were chosen so as to maximize utility, not so as to maximize fit. The predictions of the model were then compared to the participants' behaviours. The results suggest that when people learn which strategy to use through reinforcement learning, they may do so by using optimal discounting of past experience.

The remaining paper is organized as follows. The task is introduced in the next section and is followed by a description of the model, called the bounded Optimal Discounting (OD) model. Subsequently, the model results are presented, followed by a comparison between the current model predictions and those predicted by an alternative model in which the individual models use the same discounting parameter, which is called Non-optimal Discounting (ND) model .

The Task

The modeled data was acquired from the experiment reported by Howes et al. (submitted). The participants were required to copy appointment information from an email interface to a calendar. Appointments were presented in trials. On each trial, participants were asked to view various numbers of appointments on the email window one by one, ranging from 3 to 9. Since the first appointment was always at 09:00 AM and these appointments were always one hour apart and in sequence, only the names and the order they were presented need to be remembered. Once the last appointment was shown, the ‘OK’ button on the email window enabled the participants to go to the calendar window, with the email window disappearing, and copy these appointments across by typing these names in the time slots. Once they were satisfied with their copy and clicked the ‘Finish’ button, they would receive feedback about the number of appointments correctly copied and highlighted in red any slots incorrectly completed.

An important difference between the studies of Howes et al. (submitted) and of Gray et al. (2006) is that the Howes et al. study was designed with two-phases, a no-choice phase followed by a choice phase. In the no-choice phase, the *strategy* that participants adopted on each trial was assigned by the system (the number of appointments that participants were required to view before copying across was regarded as a *strategy*, ranging from 3 to 9). During this phase, they were asked to copy 100 correct appointments (only correctly copied items were counted in the target total items), and the strategies (3, 4, 5, 6, 7, 8 and 9) appeared almost evenly. The reason to do so was to force the participants to explore across the strategy space so that it allows us to empirically measure their performance over the strategies. After this phase, entering in Choice-phase, the participants were required to copy 200 appointments correctly by selecting their own preferred strategies on each trial. In addition, participants were asked to minimize the total time taken for the task and as they had to copy a target number of correct items, they were effectively asked to optimize the speed/accuracy trade-off. Therefore, the utility of each strategy was defined in term of *reward rate*, which was defined as the rate of successful copies.

The Bounded Optimal Discounting (OD) Model

The purpose of the model is to explain strategy choice on simple remembering tasks. As we have said, rather than maximizing the fit of the model to the data, a key feature of the model is that remembering strategies, and the experience discounting parameter, are chosen so as to maximize utility. The remembering strategy space consists of strategies for remembering 1 to 9 items on each visit to the calendar. The choice of the discounting parameter, named *StepSize*, has consequences for the weight given to a reward when estimating the future utility of a remembering strategy. In our model, the discounted parameter that is used to update

the trial-by-trial strategy value estimates is set so as to optimize the overall utility of the model for each individual.

Detailed Description of Optimal Discounting (OD) model

RL is concerned with learning to obtain rewards or avoid punishments by trial and error (Sutton & Barto 1998; Daw & Frank, 2009; Cohen, 2008). It has been used to understand how iterated rewards and punishments (experience) determine choice behavior in various situations. In particular, how the structure, amount, hierarchy etc. of the observed experience relate to the learning results has attracted increasing attention (Botvinick & Barto 2009). A reinforcement learning model with strategy-utility updating based on recency weighted experience is used in our analysis.

The model is defined by three parameters, $[S, R, E]$, and strategy-value estimation updating rules. S is the strategy space, $S = \{S_1, S_2, \dots, S_i, \dots, S_n\}$. The strategy taken on trial t is denoted $S(t)$. Once the strategy has been selected, the environment would give reward from the reward set R , $R \in [0, 1]$. The reward following the strategy S_i on trial t is denoted as $r_i(t)$. In this learning problem, each strategy has an expected or mean reward given that that strategy is selected, called the true (actual) *value* of this strategy. To measure the utility of the strategies trial-by-trial, the model uses estimated values acquired through experience. Specifically, on each trial t , $t \in [1, 2, \dots]$, the model updates an estimate vector, $E(t) = \{E_1(t), E_2(t), \dots, E_i(t), \dots, E_n(t)\}$, where $E_i(t)$ is the estimate of strategy value of S_i on trial t . The initial estimated value of each strategy is 0, i.e. $E_i(0) = 0$, where $i \in [1, 2, \dots, n]$. In addition, because the values of remembering strategies are non-stationary, due to practice, a discounting technique is applied to the experience when estimating the strategy-value. Specifically, as people practice a strategy they improve. This process of improvement means that for any pair of strategies i and j , the relationship between their true values at trial t , e.g. $V_i(t) > V_j(t)$, will not necessarily hold after an increment in the practice of i , the practice of j , or both. Therefore, in order to track this non-stationary learning environment, more recent experience might deserve to be weighted more heavily than temporally distant experience. Here we adopt one of the most popular ways to achieve experience discounting, called the *exponential, recency-weighted discounting*. Specifically, if a strategy has been chosen k times before, yielding rewards r_1, r_2, \dots, r_k , then the value of this strategy is estimated to be

$$E_k = \sum_{i=1}^k \left(\frac{c(1-c)^{k-i}}{\sum_{i=1}^k c(1-c)^{k-i}} \right) \times r_i \quad (1)$$

Where c ($c \in (0, 1]$) is the discounting parameter, called *StepSize*, which determines the weighting of previously received rewards. The weight given to a reward r_i depends on how many rewards previously, $k-i$, it was observed. As $(1-c)$ is always less than 1, the weight given to r_i decreases

as the number of intervening rewards increases. In fact, the weight decays exponentially according to the exponent on $1-c$. The higher the value of the *StepSize*, then the more recent rewards will contribute to the estimate relative to distant rewards. Figure 1 below gives the weight distributions of

$$f(c) = \frac{c(1-c)^{k-i}}{\sum_{i=1}^k c(1-c)^{k-i}} \quad (2)$$

with $k=8$, $i=1, 2, \dots, 8$, and five sets of c , [0.1, 0.3, 0.5, 0.7, and 0.9]. As you can see, the line of $c=0.1$ (the red one) is much more flat than the line of $c=0.9$ (the green one), which means that the relative distant rewards, like r_5, r_6 , under the $c=0.1$ model contribute more to estimate the future utility than they do under the model with $c=0.9$ in which the estimation mostly relies on two latest rewards r_7, r_8 .

In the OD model the value of the *StepSize* parameter was chosen so as to optimize utility for each individual. Specifically, given the means of the estimated strategy values above and a specific *StepSize*, then on each trial, the strategy with the highest estimate, i.e. the greedy action, is taken as the prediction of the participant's behaviour. For these predicted strategies, the model also gives predictions of their rewards. On each trial, the mean of the rewards received by the predicted strategy is regarded as the predicted reward of this strategy. Therefore, for each set of the *StepSize* we get a set of predicted strategies and rewards for each participant. We find a *StepSize* that generates maximal overall reward for each participant.

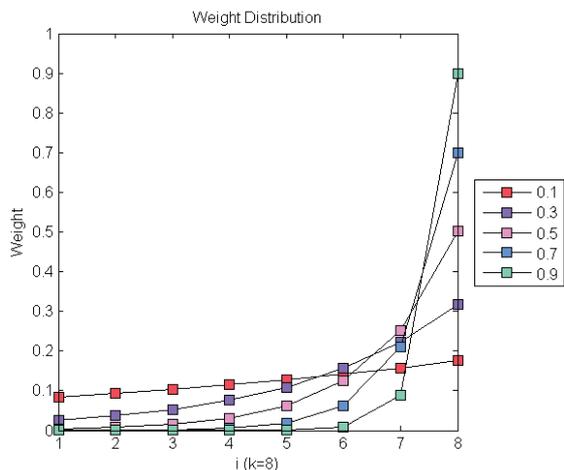


Figure 1: The weight distributions.

Alternative Model

In order to test the OD model we compared it with a model in which the *StepSize* is set to be 0.1 for all the participants. This value offers very little discounting (Figure 1). Specifically, on each trial, the values of the strategies which are selected for k times with rewards r_1, r_2, \dots, r_k , are estimated according to the equation (1) with $c=0.1$. In other words, for this model, there are two key features. First, according to the weight distribution with $c=0.1$ in Figure 1

you can see that the weights put on the experienced rewards differ only a little, which means that previous rewards almost equally contribute to the future utility estimation. Second, the same parameter value is used for all the participants. As with the OD model, the greedy action on each trial of the choice phase is predicted to be the participants' behavior, and on each trial, the mean of the rewards received by the predicted action is regarded as the predicted reward of this action. We call this model the Non-optimal Discounting (ND) model, i.e. the model with a fixed low- discounting parameter for all participants.

Predictions of the models

Both models, OD and ND, predict trial-by-trial individual participant strategy selections on the basis of the strategy-value estimates. In addition, they predict the rewards following the predicted actions, so that we could find a predicted action set that maintains the maximal expected reward. As we have said, for OD, the weights given to the rewards are adjusted by setting the discounting parameter c to a value that optimizes expected reward for each participant. For ND, the parameter is set to be 0.1 for all the participants. Comparison between OD and ND model allows us to test the assumption that people adapt the discounting parameter so that it is optimal given the constraints imposed by practice. If OD makes significantly better predictions than ND then we have evidence that participants discounted their previous experience given the expected effects of practice on strategy value.

Consequently, for both models we obtained the predicted actions and reward rates on each trial. Despite the fact that neither model is fitted to the data we expect OD to offer significantly better predictions than ND.

Results

Overall Performance over the Strategy Space

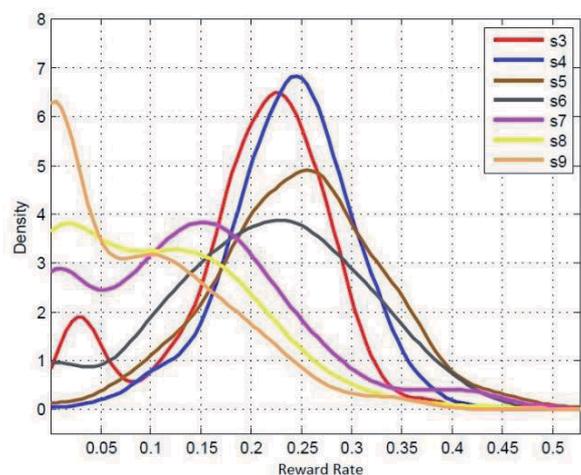


Figure 2: the probability densities of the reward rate for each strategy over all the participants.

For each participant and each trial, the following experimental data was recorded: selected strategy (one of 3, 4, 5, 6, 7, 8 or 9 items, including the strategies assigned by the system in the no-choice phase and the strategies chosen by the participants in the choice phase), the number of items correctly copied items, and the trial duration. The *reward rate* of the selected strategy is computed as the number of items correctly copied at a trial over the trial duration. Figure 2 (above) gives the overall measurement of each strategy's performance over all the 40 participants during the experiment. As shown in the figure, strategies 3, 4, 5 are the three most effective strategies across participants (Mean=5.6773, SD=1.8505, Mode=5). It is also evident that some of the strategies have bimodal densities, reflecting the low reward rates associated with error trials.

Descriptive Results

First consider the predictions of the OD model. As mentioned above, an OD model with a discounting

parameter *StepSize* that maximizes the sum of predicted rewards over the choice phase was found for each participant. In Figure 3 (below), each panel represents trial-by-trial value estimates for a participant. X-axis represents trials; Y-axis is the strategies' value estimations calculated by the OD model trial by trial. Each strategy is represented by a different colour, as shown in the legend on the right side of the figure. To the left side of the vertical black line is the no-choice phase; on the right side is the choice phase. The participant strategy on a trial is represented by a black circle (including the strategies assigned by the system in the No-Choice phase and the strategies chosen by the participants as their preferences in the Choice phase). The title of each panel includes information about the participant number and the *StepSize* found for the participant. Participants 15, 19, 7, 8 were selected to demonstrate the diversity of the individual performance. For comparison we divide the participants into three groups.

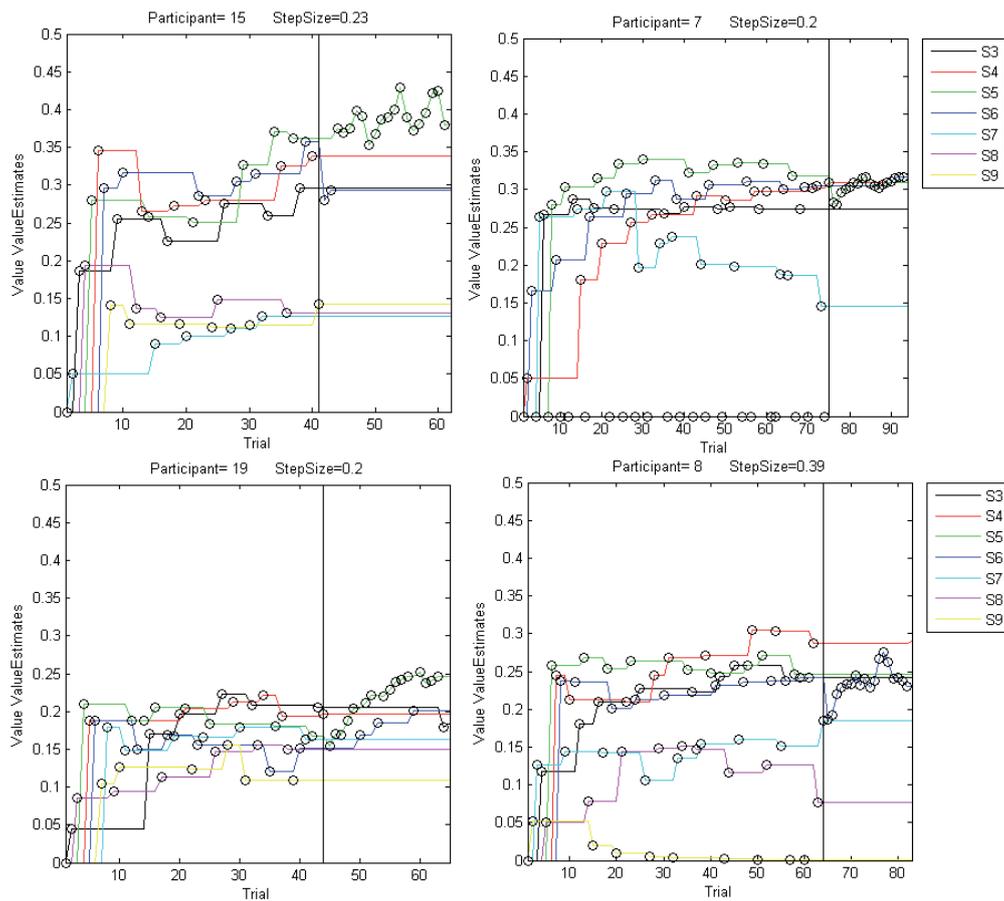


Figure 3: OD model predictions. X-axis represents trials; y-axis is the value estimates for the strategies calculated with the OD model. Each strategy is represented by a different colour, as shown in the legend on the right. To the left side of the vertical black line is the no-choice phase, on the right side is the choice phase. The selected strategy on a trial is represented by a black circle (including the strategies assigned by the system in the No-choice phase and the strategies chosen by the participants in the Choice phase). The title of each panel includes the information of the participant number and the *StepSize* found for the participant.

Group 1: *The best strategy was selected on the majority of trials in the choice phase, such as participants 21 and 14.*

Specifically, for participant 15 (top left panel), the strategy S5 became the best one (with the highest value estimate) by the end of the no-choice phase, and the participant used it on most trials in the choice phase. While for participant 19 (bottom left panel), the strategy S5 is not the best at the beginning of the choice phase, but its performance improved with practice, became best, and was chosen by the participant at the later stage of the choice phase. For the OD model, 27 of the 40 participants exhibited a pattern that was either consistent with participant 40 or 20. (*StepSize* was found between 0.03 and 0.82). For the ND model, 22 out of 40 participants behave in this way.

Group 2: *There is no clear bounded optimal strategy in most trials of the choice phase, e.g. participant 7.*

For some participants such as participant 7 (top right panel), there are several best strategies (in this case, S4, S5 and S6) with, informally, close value estimates, or it is the case that the best strategy frequently changes during the choice phase. Therefore, many strategies appear to have the highest reward and it is rational to keep exploring through the choice phase. Overall, for the OD, 8 out of 40 participants were predicted to be in this group, while 9 out of 40 for the ND model.

Group 3: *There was a clear best strategy predicted, but the participant did not end up choosing it, e.g. participant 8.*

From the beginning of the choice phase, S4 was a clear best strategy for participant 8 (bottom right), but the participant chose the strategy S6, which was unlikely to be the highest reward strategy. Overall 5 of the 40 participants behave in this pattern according to the OD model. For the ND model, 9 out of 40 participants are in this group.

Model Comparison

We computed the Root Mean Square Error (RMSE) between the strategies predicted by the model and the observed participant behaviours. The Lower RMSE, the better the model prediction. For the ND model, RMSE between predicted and observed actions in the choice phase is 1.2845, while it is 1.1539 for the OD model. In addition, we calculated RMSE between the received rewards and the predicted rewards for these two models, 0.0782 and 0.0703 for ND model and OD respectively.

We computed t-tests on the Mean Squared Errors (MSE) to determine which of these two models offered better predictions on the strategy during the choice phase. A paired right-tailed t-test between ND model and OD model indicated that the OD model, with the discounting parameter that maximises the expected reward rate is able to offer significantly better predictions of strategy choice ($t(39)=1.80$, $p=0.0396$).

Discussion

The results support the hypothesis that a model that makes bounded optimal use of internal resource (memory and

experience of reward) so as to select strategies for remembering is able to predict the majority of participant choices. In particular,

- (1) For the OD model, a discounting parameter, *StepSize*, was used to control the weights put on the rewards received by the strategies when estimating the values of the strategies for predictions of subsequent behaviour. The OD model with the *StepSize* that maximized the expected rewards for each individual participant offered a significantly better prediction of the observed data than the ND model, which weighted the received rewards with a fixed, minimal, parameter value of 0.1 to estimate the value of the strategies for all participants.
- (2) The *StepSize* that maximized the expected reward for the participants had a large range, ranging from 0.03 to 0.82. This may reflect the ability of participants to optimally adjust learning parameters to reflect meta-knowledge about the effects of their own practice on skill.

General Discussion

According to a number of studies and models, memory bounds human performance in many complex tasks, e.g. reasoning, comprehension, and learning (Cowan, 2005; Vaughan & Herrnstein, 1987). The reported study suggests that people can make bounded optimal use of memory in an everyday interactive task (copying information from email messages). In addition, people are able to strategically adjust learning parameters in response to estimates of expected reward that are non-stationary because of practice of a cognitive skill. It appears that people are able to do as well as they do on remembering tasks by selecting optimal strategies according to the cost/benefit structure of their own discounted experience of practice.

Our finding in favour of optimal strategies was not supported by the data from every individual participant. For example, participant 11 was still highly exploratory in choice phase and was not predicted by the model. However, the findings do suggest that a model that uses an optimal discounting parameter *StepSize* (OD model) does make better predictions than a model in which a fixed discounting parameter is used to predict all participants and most rewards information from the experience are used almost equally (ND model with $c=0.1$).

Further tests of the model are required to determine, for example, how well the OD model does relative to the best-fitting model, where the best fitting model adjusts *StepSize* so as to fit the data. It is inevitable that the best-fitting model will be at least as good as OD but any gap between how well the two models correspond to the data will tell us something about how much variance is unexplained by OD.

There was also evidence that some participants selected strategies that were not optimal in the early parts of the choice phase, but that with practice were improved and by the end they were generating the highest rewards. This fact is consistent with the observation that the learning environment was non-stationary because of the acquisition

of knowledge through practice. There are many studies that focus on the improvement of strategies with practice but in this paper our focus has instead been on how choices are made between strategies given that, through the effects of practice, strategies have non-stationary utility. Our starting point is the assumption that an estimate of the future utility of a strategy can be based on previous experience but that in the non-stationary environment construed by practice, it is valuable to discount the past so that more recent experience is weighted more heavily than temporally distant experience.

Conclusion

The paper provides quantitative evidence for the hypothesis that people are bounded optimal when learning to choose strategies that improve with practice. They appear to be able to manage their internal resource and learning strategies so as to maximize performance against an externally imposed payoff function.

Acknowledgments

The work was funded by a PhD scholarship awarded by the Human-Computer Interaction Centre, School of Computer Science, University of Birmingham.

References

- Anderson, M. C., Ochsner, K.N., Kuhl, B., Cooper, J., Robertson, E., Gabrieli, S.W., Glover, G.H., Gabrieli, J.D.E. (2004). Neural systems underlying the suppression of unwanted memories. *Science*, *303*, 232-235.
- Botvinick, M., Niv, Y., & Barto, A. G. (2009). Hierarchically organized behavior and its neural foundations: A reinforcement learning perspective. *Cognition*, *113*, 262-280.
- Cohen, M. X. (2008). Neurocomputational mechanisms of reinforcement-guided learning in humans: A review. *Cognitive, Affective & Behavioral Neuroscience*, *8*, 113-125.
- Cowan, N. (2005). *Working memory capacity*. Hove, East Sussex, UK: Psychology Press.
- Daw, N. D., & Frank, M. J. (2009). Reinforcement learning and higher level cognition: Introduction to special issue. *Cognition*, *113*, 259-261.
- Gray, W. D., Simms, C. R., Fu, W.-T., & Schoelles, M. J. (2006). The soft constraints hypothesis. A rational analysis approach to resource allocation for interactive behavior. *Psychological Review*, *113*, 461-482.
- Griffiths, T. L., Kemp, C., and Tenenbaum, J. B. (2008). Bayesian models of cognition, *Cambridge Handbook of Computational Cognitive Modeling*. Cambridge University Press.
- Griffiths, T. L. and Tenenbaum, J. B. (2006). Optimal predictions in everyday cognition. *Psychological Science* *17*(9), 767-773.
- Howes, A., Lewis, R.L. & Vera, A. (2009). Rational adaptation under task and processing constraints: Implications for testing theories of cognition and action. *Psychological Review*, *116*, 4, 717-751.
- Howes, A., Vera, A., Lewis, R.L. McCurdy, M. (2004). Cognitive Constraint Modeling: A formal approach to supporting reasoning about behavior. *Proceedings of the 26th Annual Meeting of the Cognitive Science Society*, Chicago.
- Howes, A., Duggan, G., Tseng, Y-C., Kalidindi, K. & Lewis, R.L. (submitted). A bounded optimal theory of short-term remembering. Paper submitted for journal publication.
- Lewis, R.L., Howes, A., Vera, A. (2004). A constraint-based approach to understanding the composition of skill. *International Conference on Cognitive Modeling*, Pittsburgh, 2004.
- Vaughan, W., Jr. & Herrnstein, R.J. (1987). Stability, melioration, and natural selection. In L. Green & J.H. Kagel (Eds.), *Advances in Behavioral Economics*, Vol. 1 (pp. 185-215). Norwood, NJ: Ablex.

Bridging the Gap between Theory and Practice of Approximate Bayesian Inference

Johan Kwisthout (kwisthou@liacs.nl)

Leiden University, Leiden Institute of Advanced Computer Science
Niels Bohrweg 1, 2333 CA Leiden, The Netherlands

Iris van Rooij (i.vanrooij@donders.ru.nl)

Radboud University Nijmegen, Donders Institute for Brain, Cognition and Behaviour
Montessorilaan 3, 6525 HR Nijmegen, The Netherlands

Abstract

In computational cognitive science, many cognitive processes seem to be successfully modeled as Bayesian computations. Yet, many such Bayesian computations has been proven to be computationally intractable (NP-hard) for unconstrained input domains, even if only an approximate solution is sought. This computational complexity result seems to be in strong contrast with the ease and speed with which humans can typically make the inferences that are modeled by Bayesian models. This contrast—between theory and practice—poses a considerable theoretical challenge for computational cognitive modelers: How can intractable Bayesian computations be transformed into computationally plausible ‘approximate’ models of human cognition? In this paper, three candidate notions of ‘approximation’ are discussed, each of which has been suggested in the cognitive science literature. We will sketch how (parameterized) computational complexity analyses can yield model variants that are tractable and which can serve as the basis of computationally plausible models of cognition.

Keywords: Bayesian inference; approximation, NP-hard; parameterized complexity theory, algorithms, computational explanation.

Introduction

Over the last decade, Bayesian modeling has become more and more important as a modeling framework in cognitive science. Many of such Bayesian models postulate that cognitive processes perform some form of Bayesian inference.¹ Examples of such models can be found in several different cognitive domains, including vision (Yuille & Kersten, 2006), language (Chater & Manning, 2006), decision making (Sloman & Hagmayer, 2006), motor planning (Wolpert & Ghahramani, 2005), eye movement control (Engbert & Krügel, 2010), and theory of mind (Baker, Saxe, & Tenenbaum, 2009; Cuijpers et al., 2006). These models often perform well at describing and predicting human behavior on small, well-structured experimental tasks. For instance, Bayesian models have

been able to successfully model human inferences about an agent’s goal in a maze-like structure given the trajectory up so far (Baker et al., 2009) and predicting the most likely next action of a co-worker in a joint action task based on observed movements (Cuijpers et al., 2006). When confronted with task situations of real-world complexity, however, it is not evident if and how these models could scale up. The reason is that Bayesian computations are known to be computationally intractable (i.e., NP-hard) for unconstrained domains (Cooper, 1990; Shimony, 1994; Park & Darwiche, 2004; Kwisthout, 2009; Kwisthout, 2011). Informally, this means that the postulated computations would simply require an unrealistic amount of time for all but small inputs if they are to operate in unconstrained domains (van Rooij, 2008).

The known NP-hardness of (unconstrained) Bayesian models of cognition implies that such models cannot serve as computationally and psychologically plausible explanations of the tasks and processes which they model. Cognitive modelers using a Bayesian approach are thus confronted with a tractability paradox: humans seem capable of performing well in situations of real-world complexity in little time and with apparently little effort; yet, our best computational cognitive models consume an unrealistic amount of time or other cognitive resources for all but overly simplified toy domains. This paradox has been recognized by both opponents and proponents of Bayesian models, and has led to considerable debate about whether or not the human mind/brain performs such (rational) probabilistic computations (Gigerenzer, Hoffrage, & Goldstein, 2008; Sanborn et al., 2010; Kruschke, 2010).

A common response from proponents of Bayesian models of cognition has been that the tractability paradox is caused by a mistaken assumption, viz., the assumption that Bayesian models necessarily aim to be *exact* models of how human minds/brains form explanations. Instead, so these researchers argue, the models postulate computations that humans compute *approximately*. For instance, the mind may approximate Bayesian computations by employing heuristics, by sampling, or by striving for merely satisfactory solutions rather than optimal solutions (Chater et al., 2006; Sanborn et al., 2010).

¹ We will use the term ‘Bayesian inference’ to denote all sorts of computations using Bayesian models, like computing posterior probabilities and finding most probable explanations, as is common in the literature; we do not restrict this term to refer to the formal INFERENCE problem in Bayesian networks.

Notwithstanding the appeal and apparent plausibility of this perspective, this standard response can at best be only part of the solution to the intractability paradox (Kwisthout, Wareham, & Van Rooij, 2011). The reason is that most—if not all—forms of approximating Bayesian inference are *for unconstrained input domains* as computationally intractable as computing Bayesian inference exactly. For instance, even though empirical results suggest that in certain specific situations a few samples are sufficient to draw reasonable conclusions from a probability distribution (Vul, Goodman, Griffiths, & Tenenbaum, 2009); making decisions based on a ‘majority vote’ using samples remains provably intractable for general input domains (Kwisthout, 2010). Similarly, while approximate inference algorithms sometimes can perform well in practice (Park, 2002), approximating or “satisficing” Bayesian inference remains intractable for general domains (Abdelbar & Hedetniemi, 1998; Kwisthout, 2011; Park et al., 2004; Roth, 1996).

The upshot of such negative computational complexity results for approximating Bayesian inference is that the assumption of ‘approximation’ by itself is insufficient to explain how Bayesian models can scale to situations of real-world complexity; and in effect, such models cannot yet claim computational plausibility as explanations of how humans make (Bayesian) inferences in everyday life (Kwisthout et al., 2011). Furthermore, by overlooking and not explicating the intrinsic complexity of approximating Bayesian inferences for certain situations, one may actually miss an opportunity to predict and explain under which circumstances humans inference can and does approximate optimal Bayesian inference and under which conditions it cannot and does not.

An alternative approach to tackle this intractability paradox is to study how the complexity of computations depends specifically on situational constraints. This approach draws among other things on the concepts and techniques from the mathematical theory of parameterized complexity theory (Downey & Fellows, 1999). Here, situational constraints are modeled by *parameters* $k_1, k_2 \dots k_m$, where each k is a property of the input. For instance, a Bayesian network may have many parameters, such as the number of cycles in the network, the maximum number of parents for any given node, the maximum length of a path between any two given nodes, the treewidth² of the network, etc.

It is known that certain NP-hard computations can be tractable for bounded ranges of such parameters. This is the case, for instance, if inferences can be computed in a time that grows non-polynomially only in the parameters and polynomial in the rest of the input size. In such situations the computations can be performed fast, even for very large inputs, provided only that the parameters take on small values in those large inputs. If similar parameterized

tractability results can be obtained for Bayesian models of cognition, this means that the models predict that humans can make fast inferences in situations that are modeled by inputs with the same bounded parameter ranges.

This approach has previously been used to successfully identify situational constraints (modeled by parameter ranges) that yield tractability for both Bayesian and non-Bayesian models of cognition in a variety of domains, such as: analogy (van Rooij et al., 2008), problem solving (Wareham, Evans, & van Rooij, 2011), similarity (Müller, van Rooij, & Wareham, 2009), action understanding (Blokpoel, Kwisthout, van der Weide, & van Rooij, 2010), and communication (van Rooij et al., 2011). For example, based on known parameterized complexity results on abduction in Bayesian networks (Bodlaender, van den Eijkhof, & Van der Gaag, 2002), van Rooij et al. (2011) derived that the inference made by an addressee to understand a communicator’s intention—as modeled by a Bayesian model—is tractable when the probability of the most probable goal is relatively high and the sender has few (instrumental) goals other than her communicative goals.

Until now, the parameterized complexity approach to dealing with intractability of cognitive models has focused on exact computations. Little research has been done, for instance, on the parameterized complexity of approximate Bayesian computations. Even the term ‘approximation’ appears to be often ill-defined. Take, for instance, again Bayesian abduction (computing the most probable explanation h out of a set of candidate explanations H , given a number of observations e , i.e., computing h such that $\Pr(H = h | e)$ is maximal). In the context of such a model ‘approximation’ could refer to finding explanations that are ‘almost as likely as’, ‘are similar to’, or ‘are likely to be’ the most probable explanation. Such ambiguity of the meaning of ‘approximation’ can lead to confusion and misunderstanding, as well as unsubstantiated claims of tractable approximability. In this paper, we aim to explicate various notions of approximate Bayesian inference in an attempt to support a more thorough study of the approximability of Bayesian computations as featured in models of cognition. Additionally, we illustrate how an approximation approach and the parameterized complexity approach can be combined to identify the situational constraints that render Bayesian models tractably approximable.

The remainder of the paper is organized as follows. We will start by defining three distinct notions of approximation. Next, we will present a case study to illustrate the utility of the proposed combined approach for one of these three notions of approximation. The conclusion of our illustration will be that approximation is neither *panacea* nor *placebo*: it is *not* a ‘magical ingredient’ that makes intractable computations tractable *by default*. Yet, it may be a ‘necessary ingredient’ for achieving tractability under certain constraints. Finally, we conclude by discussing the broader implications of our case study and

² Treewidth is a graph-theoretical concept that can loosely be described as a measure on the “localness” of connections in a graph. We refer the reader to, for instance, Bodlaender (2006) for a formal definition and more details.

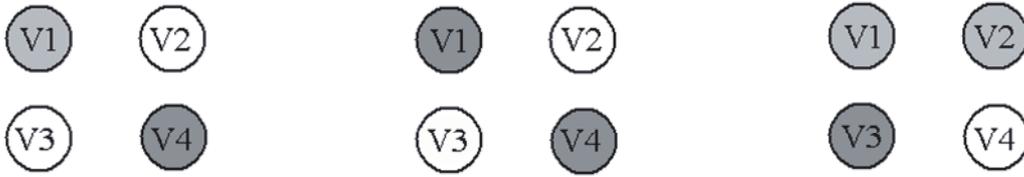


Figure 1: (left) The most probable value assignment for V1 to V4 (denoted by the light- middle- and dark-grey colors); (middle) an approximation that resembles the most probable value assignment but may have a much lower probability; (right) an approximation that may have an almost-as-high probability as, but does not resemble, the most probable value assignment

observations for dealing with the tractability paradox in Bayesian modeling in general.

Notions of approximation

In everyday language, the concept ‘roughly’ is widely used: The ratio between the circumference and diameter of a circle is *roughly* 3.14; the Sun is *roughly* 100 times as large (and also 100 times as distant from the Earth) as the Moon; there is a mountain range on Mars that looks *roughly* like a human face. We save a lot of effort if we don’t compute 98 times 102 exactly, but compute it as ‘roughly 100 times 100 = 10,000.’ When we come to see *approximation* as a synonym to *roughly computed*, in the above-sketched informal sense, the term has a pleasant association with ‘fast’, ‘easy’, and ‘without much effort’.

In computer science, however, there is a great variety on how fast and how well approximation algorithms perform for problems that are intractable to compute exactly. For example, the *knapsack* problem—in which one is to select a subset of numbers from a set of numbers so that the subset adds up to some pre-specified value—can be approximated within any desired guaranteed error margin in an amount of time that is polynomially bounded both in the input size and the error (Kellerer & Pferschy, 1999). In contrast, the *maximum clique* problem cannot be approximated at all in polynomial time, under the common assumption that $P \neq NP$ (Zuckerman, 2006). In these classic results in computer science, approximations are defined as solutions that have a *value* that is close to the optimal solution, i.e., the approximated value is within a given ratio of the value of the optimal solution. However, an approximation may also be defined as a solution that has a *structure* that is similar to the structure of the optimal solution, regardless of its value (van Rooij & Wareham, under review). The important difference between these forms of approximation can be understood as follows: a value-approximation of the *Bayesian abduction problem* can be an explanation (i.e., a value assignment to H) that is only slightly less probable than the most probable solution, but does not resemble that optimal solution. Conversely, a structure-approximation may be an explanation that *looks a lot* like the most probable explanation, but its probability may be much lower and need not even be close to optimal. Lastly, we may see approximations as solutions that are *likely to be* the optimal solution, but allow for a small margin of *expected* error. This type of approximation is common for probabilistic algorithms that are very useful for solving a certain class of

problems (e.g., primality testing), as they are much faster than deterministic algorithms yet allow the possibility of error (Motwani & Raghavan, 1995). We will denote the latter notion of approximation as *expectation-approximation*. Note that a ‘good’ value-approximation does not imply a ‘good’ structure- or expectation-approximation, or vice versa (see Figure 1).

When invoking ‘approximation’ as an explanatory tool, cognitive modelers may utilize many (combinations of) these notions of approximation. For example, models may postulate that cognitive processes compute an approximation that resembles the optimal solution for a Bayesian inference problem as well as have an almost-as-high probability value (Chater & Oaksford, 1999). Combinations of value-approximation and expectation-approximation are also common in, e.g., the well-known PAC-learning framework (Valiant, 1984) for machine learning. Consistent with the positions argued by Kwisthout et al. (2011, p. 780) and van Rooij & Wareham (under review) we submit that any claim that Bayesian models are “tractable due to approximation” should be supported by: (a) a precise definition of the notion of approximation the modeler assumes to be used; (b) in the case that the (formal) approximation problem in itself is NP-hard, a set of *problem parameters* that the modeler believes to be constrained in real-world situations (where humans perform the task well); (c) a formal proof that the chosen formal definition of the assumed approximation becomes *tractable* when the values of these parameters are so constrained; and (d) arguments or evidence that support the assumption in (b).

We will next illustrate with a case study how (a), (b), (c) and (d) can be implemented for Bayesian models using formal theoretical methods.

Case study: Most Simple Explanations

A common computational problem in Bayesian networks is inferring the best explanation out of a set of possible hypotheses, given some observations that are entered as evidence in the network. In such a network, the set of variables is typically partitioned into *explanation* variables (for which an explanation is sought), *evidence* variables (whose value is observed), and *intermediate* variables (that are neither observed, nor to be explained). This partition corresponds to the assumption that there are (many) variables that influence the resulting most probable hypothesis, but are not themselves observed. One way of dealing with such intermediate variables was proposed by

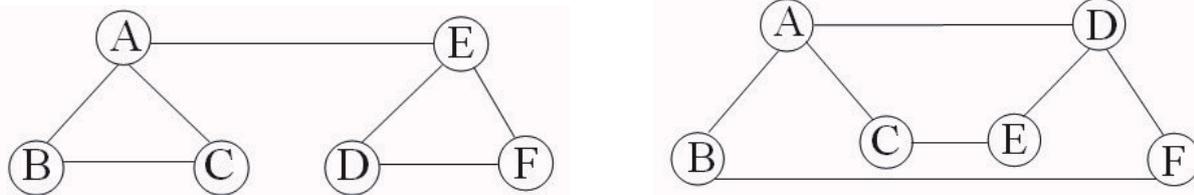


Figure 2: A network with low (left) and high (right) treewidth. Note that there is only a single connection between the clusters A,B,C and D,E,F in the left picture, whereas the connections in the right picture are much more scattered over the whole network.

Kwisthout (2010). Therein, the MOST SIMPLE EXPLANATION problem was defined as follows:

MOST SIMPLE EXPLANATION (informal)

Input: A probabilistic network, partitioned into a set of observed evidence nodes, an explanation set, and a set of intermediate variables.

Output: The joint value assignment to the explanation set that is the most probable explanation for the maximal number of instantiations of the intermediate variables.

Or more formally stated:

MOST SIMPLE EXPLANATION (formal)

Input: A probabilistic network B , partitioned into a set of evidence nodes E with a joint value assignment e , an explanation set H , and intermediate variables I .

Output: The joint value assignment h for which $\operatorname{argmax}_H \Pr(H, I = i, e) = h$ holds for the largest number of joint value i assignments to I .

Intuitively, one can think of MSE as the computational problem of finding the explanation that is most probable in the *majority of possible worlds*. Solving the MSE problem is, like many other computational problems defined over Bayesian networks, known to be intractable.³ Given this unfavorable complexity result, can MSE still serve as an adequate approximate model of abduction by humans in real-world situations? Using the four requirements introduced in the previous section we will show that the answer is *yes*: one can define an expectation-approximation algorithm for MSE that is tractable under conditions that seem to be met for human abduction.

a) Give a precise definition of the notion of approximation that is used. We use the following expectation-approximation to MSE, i.e., rather than exactly computing the solution to MSE, we want a solution that is very likely to be the MSE, but allows for a (guaranteed to be) small probability of error. In principle, such an expectation-approximation could be computed for MSE using a sampling algorithm that samples N random joint value assignments to I and casts a majority vote over the

samples. Moreover, using the so-called *Chernoff bound* (Chernoff, 1952) it can be exactly computed how many samples are required to expectation-approximate MSE to a given threshold probability. Alas, it is not possible to *tractably* expectation-approximate MSE, because the number of samples that are needed for a given degree of expectation-approximation can be exponential in the size of I .

b) As the chosen approximation is intractable in general, we define problem parameters that we hypothesize to be constrained in the ‘real world’ situations that the model should capture. A closer look at the MSE problem will reveal many problem parameters on the probability distribution and on the structure of the network. We look at three of them more closely:

1. The *treewidth* of a Bayesian network is a measure on the network structure. A formal definition of treewidth is mathematically non-trivial (see, e.g., Bodlaender, 2006, for details), but for our purposes it suffices to say that typically, if treewidth is low, the connections tend to be fairly *local* in the network. See Figure 2 for an illustrative example of this property.

2. The *cardinality* of a Bayesian network indicates the maximum number of values any variable can take; in binary variables the cardinality is two, but in principle variables can take many values.

3. The *relevancy* of the set of intermediate variables is a measure on the probability distribution in the network, and is defined as the probability that two random samples i_1 and i_2 would yield *different* most probable explanations. Informally, the intermediate variables have a low relevancy when there are only few possible worlds in which the most probable explanation deviates from the most probable explanation in the majority of worlds.

c) Give a formal proof that expectation-approximation of MSE becomes tractable when the values of these parameters are constrained. It can be shown that if both the cardinality and the treewidth of the network are small, and in addition the intermediate variables have a low relevancy, then having only a few samples already suffices to solve the MSE with a low margin of error. Consider the following algorithm from Kwisthout (2010):

³ In fact, it is NP^{PP} -hard and thus resides in the same complexity class as, e.g., the MAP (Park & Darwiche, 2004) and Parameter Tuning (Kwisthout & van der Gaag, 2008) problems in Bayesian Networks.

```

for  $n = 1$  to  $N$  do
  Choose  $i$  at random
  Determine  $h = \operatorname{argmax}_H \Pr(H, i, e)$ 
  Count the joint value assignments  $(h, i)$ 
end for
Decide upon the majority and output  $h_{maj}$ 

```

Computing $\operatorname{argmax}_H \Pr(H, i, e)$ can be done in time $O(c^{tw} \cdot n)$, where c denotes the cardinality of the network, tw the treewidth of the network, and n the number of variables of the network (Shimony, 1994). Also, according to the Chernoff bound, the number of samples N depends—for a fixed error rate—only on the probability that a randomly chosen sample actually is one of these possible worlds for which h_{maj} is the most probable explanation (i.e., depends only on the relevancy measure R) such that less samples are needed if this probability is high. We conclude that, for small c small tw , and R close to 0, this algorithm tractably expectation-approximates the MSE problem.

d) Support the hypothesized constraints in b) with arguments or empirical findings. The most contributing factor in the running time of computing $\operatorname{argmax}_H \Pr(H, i, e)$ is the treewidth of the network, as it is in the exponent. There is evidence in the machine learning literature (see e.g. Koller & Friedman, 2009) that bounding treewidth while learning Bayesian networks from data prevents overfitting of the network. As there is reason to believe human learning tends to prevent overfitting as well, the same principles may yield bounded treewidth for human belief networks. A second important factor in the running time is the relevancy of intermediate variables, i.e., the number of samples required to compute the MSE with low probability of error. While we are unaware of studies that *empirically* confirm that relevancy for human belief networks is low (i.e., number of required samples is small), there is independent theoretical research (e.g., Stewart, Chater, & Brown, 2006; Vul et al., 2009) pointing in this direction,

Conclusion

Bayesian models of cognition are evidently successful in describing and predicting many cognitive phenomena. Despite this success, one important theoretical challenge remains to date: The tractability paradox, i.e., the apparent conflict between the computational intractability of Bayesian models for general input domains on the one hand, and the ease and speed with which human perform the modeled tasks in situations of real-world complexity on the other. Often, the paradox is attempted to be solved by claiming that minds/brains approximate, rather than exactly compute, the modeled inference problems. While we agree that approximation may be part of a solution, claims of ‘approximability’ alone are in the end explanatorily unsatisfactory as they are typically too ill-defined and sometimes even provably contradicted by known complexity results (Kwisthout et al., 2011).

In this paper we presented a method for solving the tractability paradox. Importantly, our method does not solve

in one go the paradox for all Bayesian models. This is currently not possible, and may even be impossible in principle. Instead, our method describes how conditions for tractability can be studied and identified for different Bayesian models case by case. Crucial steps in our method are the following. First, one needs to decide on a relevant formal notion of approximation for the respective model. Here, we have discussed three such possible formal notions—viz., value-, structure-, and expectation-approximation—but altogether different notions or any combination of these are possible to adopt in our methodology well. Second, using techniques from computational complexity theory one investigates under which conditions the respective model can be tractably approximated in the respective sense. Last, if there are good reasons to believe that the identified tractability conditions are met in those situations where humans can perform the modeled inferences quickly, the analysis has solved the tractability paradox for this specific Bayesian model.

We illustrated the use of our methodology for a specific Bayesian model: MSE (which stands for ‘Most Simple Explanation’). We found that MSE is intractable to expectation-approximate for unconstrained Bayesian networks, but is tractable to expectation-approximate for networks of bounded treewidth and with few relevant intermediate variables. Importantly, MSE is intractable to compute exactly under those same conditions. This set of results underscores our idea that approximation is not a panacea (approximation is not enough to yield tractability for all domains), but it is not a placebo either (it plays a necessary role in yielding tractability in certain conditions). We believe that by systematically studying this interplay between approximation and constraints on input domains for more Bayesian models, these models can achieve higher levels of computational plausibility, and at the same time serve as computational-level explanations of when and why humans are good at making the rational inferences postulated by these models, and when not.

References

- Abdelbar, A.M., & Hedetniemi, S.M. (1998). Approximating MAPs for belief networks is NP-hard and other theorems. *Artificial Intelligence*, 102, 21–38.
- Baker, C.L., Saxe, R., & Tenenbaum, J.B. (2009). Action understanding as inverse planning. *Cognition*, 113, 329–349.
- Blokpoel, M., Kwisthout, J., van der Weide, T., & van Rooij, I. (2010). How action understanding can be rational, Bayesian and tractable. *Proceedings of the 32nd Annual Conference of the Cognitive Science Society*, pp. 1643–1648.
- Bodlaender, H.L. (2006). Treewidth: characterizations, applications, and computations. *Proceedings of the 32nd International Workshop on Graph-Theoretic Concepts in Computer Science*, pp. 1–14.
- Bodlaender, H.L., van den Eijkhof, F., & van der Gaag, L.C. (2002). On the complexity of the MPA problem in

- probabilistic networks. *Proceedings of the 15th European Conference on Artificial Intelligence*, pp. 675–679.
- Chater, N., & Manning, C.D. (2006). Probabilistic models of language processing and acquisition. *TRENDS in Cognitive Sciences*, 10(7), 335–344.
- Chater, N., & Oaksford, M. (1999). Ten years of the rational analysis of cognition. *Trends in Cognitive Sciences*, 3, 57–65.
- Chernoff, H. (1952). A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Annals of Mathematical Statistics*, 23(4), 493–507.
- Cooper, G.F. (1990). The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence* 42(2), 393–405.
- Cuijpers, R., Van Schie, H.T., Koppen, M., Erlhagen, W., & Bekkering, H. (2006). Goals and means in action observation: A computational approach. *Neural Networks*, 19, 311–322.
- Downey, R., & Fellows, M. (1999). *Parameterized Complexity*. Berlin: Springer.
- Engbert, R., & Krügel, A. (2010). Readers use Bayesian estimation for eye movement control. *Psychological Science*, 21(3), 366–371.
- Gigerenzer, G., Hoffrage, U., & Goldstein, D. G. (2008). Fast and frugal heuristics are plausible models of cognition: Reply to Dougherty, Franco-Watkins, and Thomas. *Psychological Review*, 115(1), 230–239.
- Kellerer, H., & Pferschy, U. (1999). A new fully polynomial time approximation scheme for the knapsack problem, *Journal of Combinatorial Optimization*, 3(1), 59–71.
- Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: Principles and techniques*. Cambridge, MA: MIT Press.
- Kruschke, J.K. (2010). Bridging levels of analysis: comment on McClelland et al. and Griffiths et al. *Trends in Cognitive Sciences*, 14(8), 344–345.
- Kwisthout, J. (2009). *The computational complexity of probabilistic networks*. Doctoral dissertation, Department of Computer Science, Utrecht University, 2009.
- Kwisthout, J. (2010). Two new notions of abduction in Bayesian networks. *Proceedings of the 22nd Benelux Conference on Artificial Intelligence*.
- Kwisthout, J. (2011). Most probable explanations in Bayesian networks: Complexity and tractability. *International Journal of Approximate Reasoning*, 52, 1452–1469.
- Kwisthout, J., & Van der Gaag, L.C. (2008). The computational complexity of sensitivity analysis and parameter tuning. *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence*, pp. 349–356.
- Kwisthout, J., Wareham, T., & van Rooij, I. (2011). Bayesian intractability is not an ailment that approximation can cure. *Cognitive Science*, 35(5), 779–784.
- Motwani, R., & Raghavan, P. (1995). *Randomized Algorithms*. New York: Cambridge University Press.
- Müller, M., van Rooij, I., & Wareham, T. (2009). Similarity as tractable transformation. *Proceedings of the 31st Annual Conference of the Cognitive Science Society*, pp. 50–55.
- Park, J.D. (2002). Using weighted MAX-SAT engines to solve MPE. *Proceedings of the Eighteenth National Conference on Artificial Intelligence*, pp. 682–687.
- Park, J.D., & Darwiche, A. (2004). Complexity results and approximation settings for MAP explanations. *Journal of Artificial Intelligence Research*, 21, 101–133.
- Roth, D. (1996). On the hardness of approximate reasoning. *Artificial Intelligence*, 82(1-2), 273–302.
- Sanborn, A.N., Griffiths, T.L., & Navarro, D.J. (2010). Rational approximations to rational models: Alternative algorithms for category learning. *Psychological Review*, 117(4), 1144–1167.
- Shimony, S.E. (1994). Finding MAPs for belief networks is NP-hard. *Artificial Intelligence*, 68(2), 399–410.
- Sloman, S.A., & Hagmayer, Y. (2006). The causal psychology of choice. *Trends in Cognitive Sciences*, 10, 407–412.
- Stewart, N., Chater, N. & Brown, G.D.A. (2006). Decision by sampling. *Cognitive Psychology*, 53, 1–26.
- Valiant, L. (1984). A theory of the learnable. *Communications of the ACM*, 27(11), 1134–1142.
- van Rooij, I. (2008). The tractable cognition thesis. *Cognitive Science*, 32(6), 939–984.
- van Rooij, I., Evans, P., Müller, M., Gedge, J. & Wareham, T. (2008). Identifying sources of intractability in cognitive models: An illustration using analogical structure mapping. *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, pp. 915–920.
- van Rooij, I., Kwisthout, J., Blokpoel, M., Szymanik, J., Wareham, T., & Toni, I. (2011). Communicating intentions: Computationally easy or difficult? *Frontiers in Human Neuroscience*, 5(52), 1–18.
- van Rooij, I., & Wareham, T. (2008). Parameterized complexity in cognitive modeling: Foundations, applications and opportunities. *The Computer Journal*, 51, 385–404.
- van Rooij, I., & Wareham, T. (2011). Intractability and approximation of optimization theories of cognition. *Under review*.
- Vul, E., Goodman, N.D., Griffiths, T.L., & Tenenbaum, J.B. (2009). One and done? Optimal decisions from very few samples. *Proceedings of the 31st Annual Conference of the Cognitive Science Society*, pp. 148–153.
- Wolpert, D.M., & Ghahramani, Z. (2005). Bayes rule in perception, action and cognition. In R.L. Gregory (Ed.), *Oxford companion to consciousness*. New York: Oxford University Press.
- Yuille, A., & Kersten, D. (2006). Vision as Bayesian inference: analysis by synthesis? *TRENDS in Cognitive Sciences*, 10(7), 301–308.
- Zuckerman, D. (2006). Linear degree extractors and the inapproximability of max clique and chromatic number, *Proceedings of the 38th ACM Symposium on Theory of Computing*, pp. 681–690.

Competence-Preserving Retention of Learned Knowledge in Soar's Working and Procedural Memories

Nate Derbinsky (nlderbin@umich.edu)

John E. Laird (laird@umich.edu)

University of Michigan, 2260 Hayward Street
Ann Arbor, MI 48109-2121 USA

Abstract

Effective management of learned knowledge is a challenge when modeling human-level behavior within complex, temporally extended tasks. This paper evaluates one approach to this problem: forgetting knowledge that is not in active use (as determined by base-level activation) and can likely be reconstructed if it becomes relevant. We apply this model for selective retention of learned knowledge to the working and procedural memories of Soar. When evaluated in simulated, robotic exploration and a competitive, multi-player game, these policies improve model reactivity and scaling while maintaining reasoning competence.

Keywords: large-scale cognitive modeling; working memory; procedural memory; cognitive architecture; Soar

Introduction

Typical cognitive models persist for short periods of time (seconds to a few minutes) and have modest learning requirements. For these models, current cognitive architectures, such as Soar (Laird, 2012) and ACT-R (Anderson et al., 2004), executing on commodity computer systems, are sufficient. However, prior work (Kennedy & Trafton, 2007) has shown that cognitive models of complex, protracted tasks can accumulate large amounts of knowledge, and that the computational performance of existing architectures degrades as a result.

This issue, where more knowledge can harm problem-solving performance, has been dubbed the *utility* problem, and has been studied in many contexts, such as explanation-based learning (Minton, 1990; Tambe et al., 1990), case-based reasoning (Smyth & Keane, 1995; Smyth & Cunningham, 1996), and language learning (Daelemans et al., 1999). Markovitch and Scott (1988) have characterized different strategies for dealing with the utility problem in terms of information filters applied at different stages in the problem-solving process. One common strategy that is relevant to cognitive modeling is selective retention, or *forgetting*, of learned knowledge. The benefit of this approach, as opposed to selective utilization, is that all available knowledge is brought to bear on problem solving, a property that is crucial for model competence in complex tasks. However, it can be challenging to devise forgetting policies that work well across a variety of problem domains, effectively balancing the task performance of cognitive models with reductions in retrieval time and storage requirements of learned knowledge.

In context of this challenge, we present two tasks where effective behavior requires that the model accumulate large

amounts of information from the environment, and where over time this learned knowledge overwhelms reasonable computational limits. In response, we present and evaluate novel policies for selective retention of learned knowledge in the working and procedural memories of Soar. These policies investigate a common hypothesis: it is rational for the architecture to forget a unit of knowledge when there is a high degree of certainty that it is not of use, as calculated by base-level activation (Anderson et al., 2004), and that it can be reconstructed in the future if it becomes relevant. We demonstrate that these task-independent policies improve model reactivity and scaling, while maintaining problem-solving competence.

Related Work

Previous cognitive-modeling research has investigated forgetting in order to account for human behavior and experimental data. As a prominent example, memory decay has long been a core commitment of the ACT-R theory (Anderson et al., 2004), as it has been shown to account for a class of memory retrieval errors (Anderson et al., 1996). Similarly, research in Soar investigated task-performance effects of forgetting short-term (Chong, 2003) and procedural (Chong, 2004) knowledge. By contrast, the motivation for and outcome of this work is to investigate the degree to which selective retention can support long-term, real-time modeling in complex tasks.

Prior work shows the potential for cognitive benefits of memory decay, such as in task-switching (Altmann & Gray, 2002) and heuristic inference (Schooler & Hertwig, 2005). In this paper, we focus on improved reactivity and scaling.

We extend prior investigations of long-term symbolic learning in Soar (Kennedy & Trafton, 2007), where the source of learning was primarily from internal problem solving. In this paper, the evaluation domains accumulate information from interaction with an external environment.

The Soar Cognitive Architecture

Soar is a cognitive architecture that has been used for developing intelligent agents and modeling human cognition. Historically, one of Soar's main strengths has been its ability to efficiently represent and bring to bear large amounts of symbolic knowledge to solve diverse problems using a variety of methods (Laird, 2012).

Figure 1 shows the structure of Soar. At the center is a symbolic working memory that represents the agent's current state. It is here that perception, goals, retrievals from

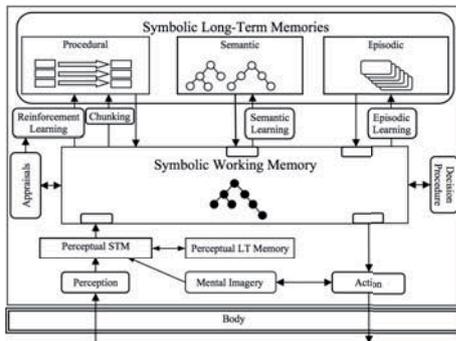


Figure 1: The Soar cognitive architecture.

long-term memory, external action directives, and structures from intermediate reasoning are jointly represented as a connected, directed graph. The primitive representational unit of knowledge in working memory is a symbolic triple (*identifier, attribute, value*), termed a *working-memory element*, or WME. The first symbol of a WME (identifier) must be an existing node in the graph, whereas the second (attribute) and third (value) symbols may be either terminal constants or non-terminal graph nodes. Multiple WMEs that share the same identifier are termed an “object,” and the set of individual WMEs sharing that identifier are termed “augmentations” of that object.

Procedural memory stores the agent’s knowledge of when and how to perform actions, both internal, such as querying long-term declarative memories, and external, such as controlling robotic actuators. Knowledge in this memory is represented as if-then rules. The conditions of rules test patterns in working memory and the actions of rules add and/or remove working-memory elements. Soar makes use of the Rete algorithm for efficient rule matching (Forgy, 1982) and retrieval time scales to large stores of procedural knowledge (Doorenbos, 1995). However, the Rete algorithm is known to scale linearly with the number of elements in working memory, a computational issue that motivates maintaining a relatively small working memory.

Soar learns procedural knowledge via chunking (Laird et al., 1986) and reinforcement learning (RL; Nason & Laird, 2005) mechanisms. Chunking *creates new* productions: it converts deliberate subgoal processing into reactive rules by compiling over production-firing traces, a form of explanation-based learning (EBL). If subgoal processing does not interact with the environment, the chunked rule is redundant with existing knowledge and serves to improve performance by reducing deliberate processing. However, memory usage in Soar scales linearly with the number of rules, typically at a rate of 1-5 KB/rule, which motivates forgetting of under-utilized productions.

Reinforcement learning incrementally *tunes existing* production actions: it updates the expectation of action utility, with respect to a subset of state (represented in rule conditions) and an environmental or intrinsic reward signal. A production that can be updated by the RL mechanism (termed in *RL rule*) must satisfy a few simple criteria related to its actions, and is thus distinguishable from other rules.

This distinction is relevant to forgetting productions. When an RL rule that was learned via chunking is updated, that rule is no longer redundant with the knowledge that led to its creation, as it now incorporates information from environmental interaction that was not captured in the original subgoal processing.

Soar incorporates two long-term declarative memories, semantic and episodic (Derbinsky & Laird, 2010). Semantic memory stores working-memory objects, independent of overall working-memory connectivity (Derbinsky, Laird, & Smith, 2010), and episodic memory incrementally encodes and temporally indexes snapshots of working memory, resulting in an autobiographical history of agent experience (Derbinsky & Laird, 2009). Agents retrieve knowledge from one of these memory systems by constructing a symbolic cue in working memory; the intended memory system then interprets the cue, searches its store for the best matching memory, and if it finds a match, reconstructs the associated knowledge in working memory. For episodic memory, the time to reconstruct knowledge depends on the size of working memory at the time of encoding, another motivation for a concise agent state.

Agent reasoning in Soar consists of a sequence of decisions, where the aim of each decision is to select and apply an *operator* in service of the agent’s goal(s). The primitive *decision cycle* consists of the following phases: encode perceptual input; fire rules to elaborate agent state, as well as propose and evaluate operators; select an operator; fire rules that apply the operator; and then process output directives and retrievals from long-term memory. Unlike ACT-R, multiple rules may fire in parallel during a single phase. The time to execute the decision cycle, which primarily depends on the speed with which the architecture can match rules and retrieve knowledge from episodic and semantic memories, determines agent reactivity. We have found that 50 msec. is an acceptable upper bound on this response time across numerous domains, including robotics, video games, and human-computer interaction (HCI) tasks.

There are two types of persistence for working-memory elements added as the result of rule firing. Rules that fire to apply a selected operator create *operator-supported* structures. These WMEs will persist in working memory until deliberately removed. In contrast, rules that do not test a selected operator create *instantiation-supported* structures, which persist only as long as the rules that created them match. This distinction is relevant to forgetting WMEs.

As evident in Figure 1, Soar has additional memories and processing modules; however, they are not pertinent to this paper and are not discussed further.

Selective Retention in Working Memory

The core intuition of our working-memory retention policy is to remove the augmentations of objects that are not actively in use and that the model can later reconstruct from long-term semantic memory, if they become relevant. We characterize WME usage via the base-level activation model (BLA; Anderson et al., 2004), which estimates future

usefulness of memory based upon prior usage. The primary activation event for a working-memory element is the firing of a rule that tests or creates that WME. Also, when a rule first adds an element to working memory, the activation of the new WME is initialized to reflect the aggregate activation of the set of WMEs responsible for its creation. The base-level activation of a WME is computed as:

$$A = \ln \left(\sum_{j=1}^n t_j^{-d} \right)$$

where n is the number of memory activations, t_j is the time since the j th activation, and d is a free decay parameter. For computational efficiency, history size is bounded: each working-memory element maintains a history of at most the c most recent activations and the activation calculation is supplemented by an approximation of the more distant past (Petrov, 2006). This model of activation sources, events, and decay is task independent.

At the end of each decision cycle, Soar removes from working memory each element that satisfies all of the following requirements, with respect to τ , a static, architectural threshold parameter:

R1. The WME was not encoded directly from perception.

R2. The WME is operator-supported.

R3. The activation level of the WME is less than τ .

R4. The WME augments an object, o , in semantic memory.

R5. The activation of all augmentations of o are less than τ .

We adopted requirements R1-R3 from Nuxoll, Laird, and James (2004), whereas R4 and R5 are novel. Requirement R1 distinguishes between the decay of representations of perception, and any dynamics that may occur with actual sensors, such as refresh rate, fatigue, noise, or damage. Requirement R2 is a conceptual optimization: as operator-supported WMEs are persistent, while instantiation-supported structures are direct entailments, if we properly manage the former, the latter are handled automatically. This means that if we properly remove operator-supported WMEs, any instantiation-supported structures that depend on them will also be removed, and thus our mechanism only manages operator-supported structures. The concept of a fixed lower bound on activation, as defined by R3, was adopted from activation limits in ACT-R (Anderson et al., 1996), and dictates that working-memory elements will decay in a task-independent fashion as their use for reasoning becomes less recent/frequent.

Requirement R4 dictates that our mechanism only removes elements from working memory that can be reconstructed from semantic memory. From the perspective of cognitive modeling, this constraint on decay resembles a working memory that is in part an activated subset of long-term memory (Jonides et al., 2008). Functionally, requirement R4 serves to balance the degree of working-memory decay with support for sound reasoning. Knowledge in Soar’s semantic memory is persistent, though may change over time. Depending on the task and the model’s knowledge-management strategies, it is possible that any removed knowledge may be recovered via

deliberate reconstruction from semantic memory. Additionally, knowledge that is not in semantic memory can persist indefinitely to support model reasoning.

Requirement R5 supplements R4 by providing partial support for the *closed-world assumption*. R5 dictates that either all object augmentations are removed, or none. This policy leads to an object-oriented representation whereby procedural knowledge can distinguish between objects that have been cleared, and thus have no augmentations, and those that simply are not augmented with a particular feature or relation. R5 makes an explicit tradeoff, weighting more heavily model competence at the expense of the speed of working-memory decay. This requirement resembles the declarative module of ACT-R, where activation is associated with each chunk and not individual slot values.

Empirical Evaluation

We extended an existing system where Soar controls a simulated mobile robot (Laird, Derbinsky, & Voigt, 2011). Our evaluation uses a simulation instead of a real robot because of the practical difficulties in running numerous, long experiments in large physical spaces. However, the simulation is quite accurate and the Soar rules (and architecture) used in the simulation are exactly the same as the rules used to control the real robot.

The robot’s task is to visit every room on the third floor of the Computer Science and Engineering building at the University of Michigan. For this task, the robot visits over 100 rooms and takes about 1 hour of real time. During exploration, it incrementally builds an internal topographic map, which, when completed, requires over 10,000 WMEs to represent and store. In addition to storing information, the model reasons about and plans using the map in order to find efficient paths for moving to distant rooms it has sensed but not visited. The model uses episodic memory to recall objects and other task-relevant features during exploration.

In our experiments, we aggregate working-memory size and maximum decision time for each 10 seconds of elapsed time, all of which is performed on an Intel i7 2.8GHz CPU, running Soar v9.3.1. Because each experimental run takes 1 hour, we did not duplicate our experiments sufficiently to establish statistical significance and the results we present are from individual experimental runs. However, we found qualitative consistency across our runs, such that the variance between runs is small as compared to the trends we focus on below.

We make use of the same model for all experiments, but modify small amounts of procedural knowledge and change architectural parameters, as described here. The baseline model (A0) maintains all declarative map information both in Soar’s working and semantic memories. A slight modification to this baseline (A1) includes hand-coded rules to prune away rooms in working memory that are not required for immediate reasoning or planning. The experimental model (A2) makes use of our working-memory retention policy and we explored different values of the base-level decay rate ($c=10$ and $\tau=2$ for all models).

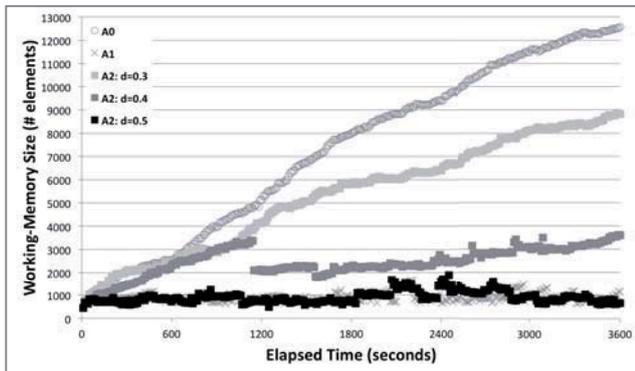


Figure 2: Model working-memory size comparison.

Figure 2 compares working-memory size between conditions A0, A1, and A2 over the duration of the experiment. We note first the major difference in working-memory size between A0 and A1 after one hour, when the working memory of A1 contains more than 11,000 fewer elements, more than 90% less than A0. We also find that the greater the decay-rate parameter for A2, the smaller the working-memory size, where a value of 0.5 qualitatively tracks A1. This finding suggests that our policy, with an appropriate decay, keeps working-memory size comparable to that maintained by hand-coded rules.

Figure 3 compares maximum decision-cycle time in msec., between conditions A0, A1, and A2 as the simulation progresses. The dominant cost reflected by this data is time to reconstruct prior episodes that are retrieved from episodic memory. We see a growing difference in time between A0 and A2 as working memory is more aggressively managed (i.e. greater decay rate), demonstrating that episodic reconstruction, which scales with the size of working memory at the time of episodic encoding, benefits from selective retention. We also find that with a decay rate of 0.5, our mechanism performs comparably to A1. We note that without sufficient working-memory management (A0; A2 with decay rate 0.3), episodic-memory retrievals are not tenable for a model that must reason with this amount of acquired information, as the maximum required processing time exceeds the reactivity threshold of 50 msec.

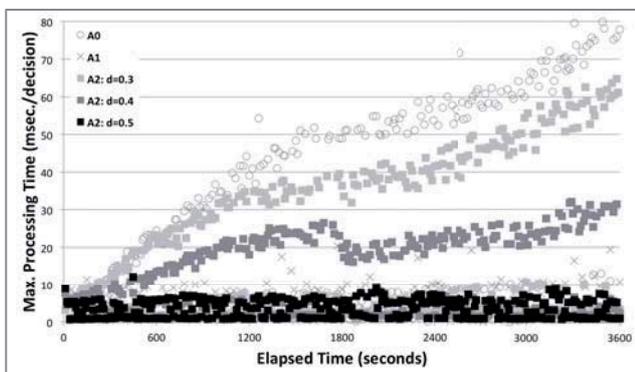


Figure 3: Model maximum decision time comparison.

Discussion

It is possible to write rules that prune Soar’s working memory; however, this task-specific knowledge is difficult to encode and learn, and interrupts deliberate processing.

In this work, we presented and evaluated a novel approach that utilizes a memory hierarchy to bound working-memory size while maintaining sound reasoning. This approach assumes that the amount of knowledge required for immediate reasoning is small relative to the overall amount of knowledge accumulated by the model. Under this assumption, as demonstrated in the robotic evaluation task, our policy scales even as learned knowledge grows large over long trials. We note that since Soar’s semantic memory can change over time and is independent of working memory, our selective-retention policy does admit a class of reasoning error wherein the contents of semantic memory are changed so as to be inconsistent with decayed WMEs. However, this corruption requires deliberate reasoning in a relatively small time window and has not arisen in our models. While the model completed this task for all conditions reported here, at larger decay rates (≥ 0.6) the model thrashed because map information was not held in working memory long enough to complete deep look-ahead planning. This suggests additional research is needed on either adaptive decay-rate settings or planning approaches that are robust in the face of memory decay.

Selective Retention in Procedural Memory

The intuition of our procedural-memory retention policy is to remove productions that are not actively used and that the model can later reconstruct via deliberate subgoal reasoning, if they become relevant. We utilize the base-level activation model to summarize the history of rule firing.

At the end of each decision cycle, Soar removes from procedural memory each rule that satisfies all of the following requirements, with respect to parameter τ :

- R1. The rule was learned via chunking.
- R2. The rule is not actively firing.
- R3. The activation level of the rule is less than τ .
- R4. The rule has not been updated by RL.

We adopted R1-R3 from Chong (2004), whereas R4 is novel. Chong was modeling human skill decay, and did not delete productions, so as to not lose each rule’s activation history. Instead, decayed rules were prevented from firing, similar to below-utility-threshold rules in ACT-R. R1 is a practical consideration to distinguish learned knowledge from “innate” rules developed by the modeler, which, if modified, would likely break the model. R2 recognizes that matched rules are in active use and thus should not be forgotten. R3 dictates that rules will decay in a task-independent fashion as their use for reasoning becomes less recent/frequent. We note that for fixed parameters (d and τ) and a single activation, the BLA model is equivalent to the use-gap heuristic of Kennedy and Trafton (2007). However, the time between sequential rule firings ignores firing frequency, which the BLA model incorporates.

Requirement R4 attempts to retain only those rules that the model cannot regenerate via chunking, a process that compiles existing knowledge applied in subgoal reasoning. Chunked rules that have been updated by RL encode expected utility information, which is not captured by other learning mechanisms. Because this information is difficult, if not impossible, to reconstruct, these rules are retained.

Empirical Evaluation

We extended an existing system (Laird et al., 2011) where Soar plays Liar’s Dice, a multi-player game of chance. The rules of the game are numerous and complex, yielding a task that has rampant uncertainty and a large state space (millions-to-billions of relevant states for games of 2-4 players). Prior work has shown that RL allows Soar models to significantly improve performance after playing a few thousand games. However, this involves learning large numbers of RL rules to represent the state space.

The model we use for all experiments learns two classes of rules: RL rules, which capture expected action utility, and symbolic game heuristics. Our experimental baseline (B0) does not include selective retention. The first experimental modification (B1) implements our selective-retention policy, but does not enforce requirement R4 and is thereby comparable to prior work (Kennedy & Trafton, 2007; Chong, 2004). The second modification (B2) fully implements our policy. We experiment with a range of representative decay rates, including 0.999, where rules not immediately updated by RL are deleted ($c=10, \tau=-2$ for all).

We alternated 1,000 2-player games of training then testing, each against a non-learning version of the model. After each testing session, we recorded maximum memory usage (Mac OS v10.7.3; dominated, in this task, by procedural memory), task performance (% games won), and average decisions/task action. We do not report maximum decision time, as this was below 6 msec. for all conditions (Intel i7 2.8GHz CPU, Soar v9.3.1). We collected data for all conditions in at least three independent trials of 40,000 games. For conditions that used selective retention, we were able to gather more data in parallel, due to reduced memory consumption (six trials for $d=0.35$, seven for remaining).

Figure 4 presents average memory growth, in megabytes, as the model trains, where the error bars represent ± 1 standard deviation. For all models, the memory growth of games 1-10K follows a power law ($r^2 \geq 0.96$), whereas for

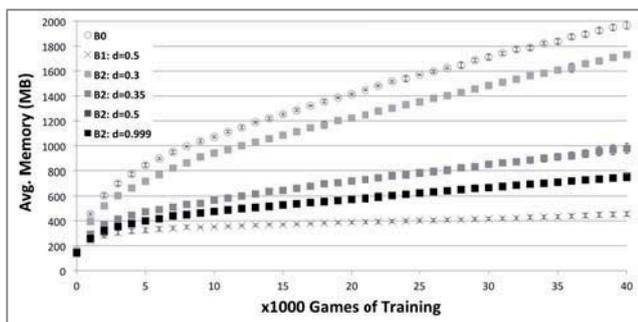


Figure 4. Avg. memory usage ± 1 std. dev. vs. games played.

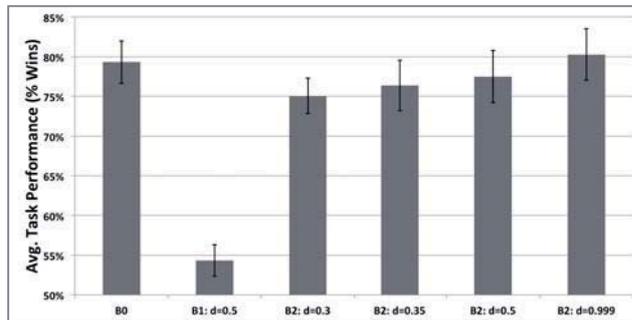


Figure 5. Avg. task performance ± 1 std. dev.

11-40K, growth is linear ($r^2 \geq 0.99$). These plots indicate that memory usage for the baseline (B0) and the slowly decaying model (B2, $d=0.3$) is much greater, and faster growing, than models that more aggressively decay. It also shows that there is a diminishing benefit from faster decay (e.g. $d=0.5$ and 0.999 for B2 are indistinguishable).

Figure 5 presents average task performance after 1,000 games of training, where the error bars represent ± 1 standard deviation. This data shows that given the inherent stochasticity of the task, there is little, if any, difference between the performance of the baseline (B0) and decay levels of B2. However, by comparing B0 and B2 to B1, it is clear that without R4, the model suffers a dramatic loss of task competence. For clarity, the model begins by playing a non-learning copy of itself and learns from experience with each training session. While the B0 and B2 models improve from winning 50% of games to 75-80%, the B1 model improves to below 55%. We conclude that a selective-retention policy that only incorporates production-firing history (e.g. Chong, 2004; Kennedy & Trafton, 2007) will negatively impact performance in tasks that involve informative interaction with an external environment. Our policy incorporates both rule-firing history and rule reconstruction, and thus retains this source of feedback.

Finally, Figure 6 presents average number of decisions for the model to take an action in the game after training for 10,000 games. In prior work (e.g. Kennedy & Trafton, 2007), this value was a major performance metric, as it reflected the primary reason for learning new rules. In this work, each decision takes very little time, and so the number of decisions to choose an action is not as crucial to task performance as the selected action. However, these data show that there exists a space of decay values (e.g. $d=0.35$) in which memory usage is relatively low and grows slowly (Figure 4), task performance is relatively high (Figure 5), and the model makes decisions relatively quickly (Figure 6).

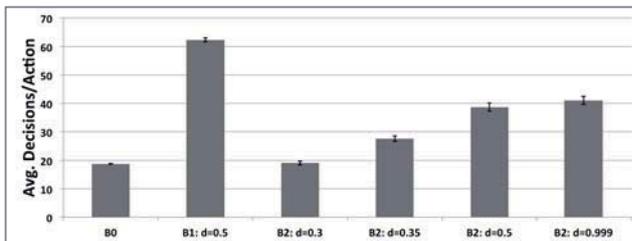


Figure 6. Avg. decisions/task action ± 1 std. dev.

Discussion

This work contributes evidence that we can develop models that improve using RL in tasks with large state spaces. Currently, it is typical to explicitly represent the entire state space, which is not feasible in complex problems. Instead, Soar learns rules to represent only those portions of the space it experiences, and our policy retains only those rules that include feedback from environmental reward. Future work needs to validate this approach in other domains.

Concluding Remarks

This paper presents and evaluates two policies for effective retention of learned knowledge from complex environments. While forgetting mechanisms are common in cognitive modeling, this work pursues this line of research for functional reasons: improving computational-resource usage while maintaining reasoning competence. We have presented compelling results from applying these policies in two complex, temporally extended tasks, but there is additional work to evaluate these policies, and their parameters, across a wider variety of problem domains.

This paper does not address the computational challenges associated with efficiently implementing these policies. Derbinsky and Laird (2012) present and evaluate algorithms for implementing forgetting via base-level activation.

Acknowledgments

We acknowledge the funding support of the Air Force Office of Scientific Research, contract FA2386-10-1-4127.

References

- Altmann, E., Gray, W. (2002). Forgetting to Remember: The Functional Relationship of Decay and Interference. *Psychological Science*, 13 (1), 27-33.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., Qin, Y. (2004). An Integrated Theory of the Mind. *Psychological Review*, 111 (4), 1036-1060.
- Anderson, J. R., Reder, L., Lebiere, C. (1996). Working Memory: Activation Limitations on Retrieval. *Cognitive Psychology*, 30, 221-256.
- Chong, R. (2003). The Addition of an Activation and Decay Mechanism to the Soar Architecture. *Proc. of the 5th Intl. Conf. on Cognitive Modeling* (pp. 45-50).
- Chong, R. (2004). Architectural Explorations for Modeling Procedural Skill Decay. *Proc. of the 6th Intl. Conf. on Cognitive Modeling*.
- Daelemans, W., Van Den Bosch, A., Zavrel, J. (1999). Forgetting Exceptions is Harmful in Language Learning. *Machine Learning*, 34, (pp. 11-41).
- Derbinsky, N., Laird, J. E. (2009). Efficiently Implementing Episodic Memory. *Proc. of the 8th Intl. Conf. on Case-Based Reasoning* (pp. 403-417).
- Derbinsky, N., Laird, J. E. (2010). Extending Soar with Dissociated Symbolic Memories. *Symposium on Human Memory for Artificial Agents, AISB* (pp. 31-37).
- Derbinsky, N., Laird, J. E. (2012). Computationally Efficient Forgetting via Base-Level Activation. *Proc. of the 11th Intl. Conf. on Cognitive Modeling*.
- Derbinsky, N., Laird, J. E., Smith, B. (2010). Towards Efficiently Supporting Large Symbolic Declarative Memories. *Proc. of the 10th Intl. Conf. on Cognitive Modeling* (pp. 49-54).
- Doorenbos, R. B. (1995). Production Matching for Large Learning Systems. Ph.D. Diss., Computer Science Dept. Carnegie Mellon, Pittsburgh, PA.
- Forgy, C. L. (1982). Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem. *Artificial Intelligence*, 19 (1), 17-37.
- Kennedy, W. G., Trafton, J. G. (2007). Long-term Symbolic Learning. *Cognitive Systems Research*, 8, 237-247.
- Laird, J. E. (2012). *The Soar Cognitive Architecture*, MIT Press.
- Laird, J. E., Derbinsky, N., Tinkerhess, M. (2011). A Case Study in Integrating Probabilistic Decision Making and Learning in a Symbolic Cognitive Architecture: Soar Plays Dice. *Papers from the 2011 Fall Symposium Series: Advances in Cognitive Systems* (pp. 162-169).
- Laird, J. E., Derbinsky, N., Voigt, J. R. (2011). Performance Evaluation of Declarative Memory Systems in Soar. *Proc. of the 20th Behavior Representation in Modeling & Simulation Conf.* (pp. 33-40).
- Laird, J. E., Rosenbloom, P. S., Newell, A. (1986). Chunking in Soar: The Anatomy of a General Learning Mechanism. *Machine Learning*, 1 (1), 11-46.
- Markovitch, S., Scott, P. D. (1988). The Role of Forgetting in Learning. *Proc. of the 5th Intl. Conf. on Machine Learning* (pp. 459-465).
- Minton, S. (1990). Qualitative Results Concerning the Utility of Explanation-Based Learning. *Artificial Intelligence*, 42, 363-391.
- Nason, S., Laird, J. E. (2005). Soar-RL: Integrating Reinforcement Learning with Soar. *Cognitive Systems Research*, 6 (1), 51-59.
- Nuxoll, A., Laird, J. E., James, M. (2004). Comprehensive Working Memory Activation in Soar. *Proc. of the 6th Intl. Conf. on Cognitive Modeling* (pp. 226-230).
- Petrov, A. (2006). Computationally Efficient Approximation of the Base-Level Learning Equation in ACT-R. *Proc. of the 7th Intl. Conf. on Cognitive Modeling* (pp. 391-392).
- Schooler, L., Hertwig, R. (2005). How Forgetting Aids Inference. *Psychological Review*, 112 (3), 610-628.
- Smyth, B., Cunningham, P. (1996). The Utility Problem Analysed: A Case-Based Reasoning Perspective. *LNCS*, 1168, 392-399.
- Smyth, B., Keane, M. T. (1995). Remembering to Forget: A Competence-Preserving Case Deletion Policy for Case-Based Reasoning Systems. *Proc. of the 13th Intl. Joint Conf. on Artificial Intelligence* (pp. 377-382).
- Tambe, N., Newell, A., Rosenbloom, P. S. (1990). The Problem of Expensive Chunks and its Solution by Restricting Expressiveness. *Machine Learning*, 5, 299-349.

Pre-attentive and Attentive Vision Module

Enkhbold Nyamsuren (e.nyamsuren@rug.nl)

Niels A. Taatgen (n.a.taatgen@rug.nl)

Department of Artificial Intelligence, University of Groningen,
Nijenborgh 9, 9747 AG Groningen, Netherlands

Abstract

This paper introduces a new vision module, called PAAV, developed for the cognitive architecture ACT-R. Unlike ACT-R's default vision module that was originally developed for top-down perception only, PAAV was designed to model a wide range of tasks, such as visual search and scene viewing, where pre-attentive bottom-up processes are essential for the validity of a model. PAAV builds on attentive components of the default vision module and incorporates greater support for modeling pre-attentive components of human vision. The module design incorporates the best practices from existing models of vision. The validity of the module was tested on three different tasks.

Keywords: vision; iconic memory; cognitive architecture; ACT-R.

Introduction

This paper introduces a general purpose vision module called PAAV, which stands for Pre-attentive And Attentive Vision. As the name suggests, the new module incorporates a greater support for bottom-up visual components that are considered pre-attentive in nature, such as multiple feature dimensions to describe visual objects, peripheral vision with differential acuity, iconic visual memory and a decision threshold. The module was developed as an integral part of ACT-R cognitive architecture (Anderson, 2007) that provides a necessary top-down, attentive layer. By being part of ACT-R, PAAV should be able to model wide range of tasks where both top-down and bottom-up visual guidances are important. ACT-R already has a default vision module and a few extensions for it. However they have drawbacks that PAAV is aimed to solve.

ACT-R's default vision module can be described in terms of a *visicon* and two buffers: *visual-location* and *visual*. *Visual-location* and *visual* buffers essentially represent WHERE and WHAT components of a visual system. The *visicon* represents the visual scene containing visual objects with which an ACT-R model can interact. The *visicon* is considered to be a part of the environment (a monitor screen) rather than part of the model. A model can send a WHERE request to the *visual-location* buffer to find the location in the *visicon* of a potential visual object to encode. Within this request, the model can specify criteria for visual object such as its kind, color, coordinates or size. Given this request vision module randomly chooses one of the visual objects from the *visicon* that exactly matches the given criteria and puts its location information in the *visual-location* buffer. This entire process is instantaneous with no

time cost. Next, model can send a WHAT request to the *visual* buffer to encode the object at the chosen location of *visicon*. A WHAT request assumes fixed execution times for both saccade and encoding that in total require 85 ms.

EMMA (Salvucci, 2001) is arguably the most used extension to ACT-R's default vision module. EMMA explicitly models saccades including preparation and execution times, path generation and variable landing points. However, EMMA's major contribution is in its ability to model covert attention shifts through variable encoding time dependent on visual object's frequency and eccentricity.

The disadvantage of the default vision module and EMMA is in their optimization toward tasks that involve reading or working with items of a user interface. Those are the tasks with relatively a simple visual environment where bottom-up perceptual processes can be ignored without sacrificing model's plausibility and performance. However, ACT-R's vision module is not suitable for tasks where visual stimuli are described with multiple feature dimensions. Such tasks often require theories of scene perception and visual search that are not part of current vision module. The issue is more pressing if one considers the importance of embodied cognition (e.g., Clark, 1997) in problem-solving tasks (Nyamsuren & Taatgen, 2011) and in everyday human activities in general (Land, Mennie & Rusted, 1999). Embodied cognition assumes that cognitive control is not purely goal based, but it is also driven perceptually. The simplest example of it is an interference of the salient feature during the task (Theeuwes, 1992). When subjects are asked to look at the scene they tend to look at the most salient parts first. Those salient parts of the scene can interfere with task even if subjects are explicitly asked to not to look at them.

Architecture of PAAV Module

Feature dimensions

In PAAV every visual object can be characterized by five basic features: color, shape, shading, orientation and size. The features are chosen because of their pop-out nature and importance in guiding visual attention (Wolfe & Horowitz, 2004). Each of those features can have a wide range of values, such as, red and green for color; oval and rectangle for shape and etc. Currently, PAAV does not support modeler specified custom features. However, it is included as a future implementation milestone.

Peripheral Vision

The current implementation of ACT-R's vision assumes that everything in a visicon is visible to the vision module and consecutively available for information processing. However, human vision is limited in what it can see, especially in the extra-foveal region (Rayner, 1998). PAAV introduces limitations on visibility by assuming that a visual object is only visible if at least one of five features of that object is visible. Visibility of a feature is calculated with an acuity function. We have adopted a modified version of the psychophysical acuity function proposed by Kieras (2010). Kieras' original acuity function states that for an object's feature to be visible the object's angular size s , with some Gaussian noise added to it, must exceed a threshold calculated as a function of eccentricity e :

$$\begin{aligned} \text{threshold} &= ae^2 + be + c \\ P(\text{available}) &= P(s + X > \text{threshold}) \\ X &\sim N(0, vs) \end{aligned}$$

The free parameters a , b , c and v are to be adjusted for each particular feature. The function works quite well for modeling differential acuity of features. However, the quadratic form in the function makes it less suitable when the object size is particularly small. For example, in their feature search experiment for color, Treisman and Gelade (1980) used visual stimuli of $0.8^\circ \times 0.6^\circ$ in size scattered over area of $14^\circ \times 8^\circ$. This feature search experiment cannot be replicated with the above acuity function for color unless parameter a is assigned an extremely low value that is well below the 0.035 used by Kieras (2010).

PAAV uses a modified version of the acuity function to mitigate issue above:

$$\begin{aligned} \text{threshold} &= ae^2 - be \\ P(\text{available}) &= P(s > \text{threshold}) \end{aligned}$$

The constant c has been removed since it has no significant influence when object size is reasonably large and too much influence when object size is quite small. Similarly, the Gaussian noise has been removed because of its tendency to introduce too much or too little acuity variation depending on the object size. Next, the coefficient b has an opposite sign. It results in less steeper increase in threshold when an eccentricity increases. It also removes the necessity of giving unreasonably small value to coefficient a when object size is small. The free parameter a has been refitted again to 0.035 and 0.1 for color and shape respectively. The parameter b has been fitted to 0.601 for both color and shape. We are still in process of refitting parameters for the rest of the features.

Iconic Visual Memory

Everything PAAV perceives from the *visicon* is stored in iconic memory. Visual features of every object visible via peripheral vision are stored in this memory. As such, the content of iconic memory is not necessarily a complete or even a consistent representation of the objects in the *visicon*.

Information in iconic memory is not treated as consciously perceived visual properties. It is rather perceived as bottom-up visual stimuli on which bottom-up processes can operate. Iconic memory is trans-saccade persistent. Items in iconic memory are persistent for a short

duration of time if they are not visible through peripheral vision anymore. This persistence time is currently set to 4 s determined by Kieras (2009) to be a lower bound for a visual memory.

Iconic memory is a model's internal representation of a *visicon*, otherwise visual scene. As such, all WHERE requests are handled with respect to the content of iconic memory via a newly defined *abstract-location* buffer. A request may include desired criteria including any of the five feature dimensions or location.

Visual Activation

Each visual object in iconic memory is assigned an activation value. The location of the visual object with the highest activation value is returned upon a WHERE request. The activation value is calculated as a sum of bottom-up and top-down activation values. It is adapted from the concept of an activation map used by Wolfe (2007) in his model of a visual search.

Bottom-up activation The bottom-up activation for a visual object i is calculated based on its contrast to all other objects in iconic memory with respect to each feature dimension k :

$$BA_i = \sum_j \sum_k \frac{\text{dissim}(v_{ik}, v_{jk})}{\sqrt{d_{ij}}}$$

The $\text{dissim}(v_{ik}, v_{jk})$ is the dissimilarity score of two feature values of the same dimension. It is a simplification of a bottom-up activation based on the difference in channel responses used in Guided Search 4.0 (Wolfe, 2007). If two values are the same then $\text{dissim}(v_{ik}, v_{jk})=0$, otherwise $\text{dissim}(v_{ik}, v_{jk})=1$. The dissimilarity is weighted by a square root of a linear distance d_{ij} between two objects. Thus the objects farther away contribute less to a contrast-based saliency of the visual object i than the objects closest to it.

Top-down activation In a WHERE request a model can provide feature values as desired criteria for the next visual object to be located. Those feature criteria are used to calculate the top-down activation value for each visual object in iconic memory. Given k feature criteria the top-down activation for visual object i is calculated as:

$$TA_i = \sum_k \text{sim}(f_{ik}, f_k)$$

$\text{sim}(f_{ik}, f_k)$ is a similarity score of the feature value f_k in WHERE request to a value f_{ik} with the same feature dimension in visual object i . This similarity score is 1 for an exact match and 0 for a mismatch. If the value f_{ik} is not accessible from iconic memory then the similarity score is considered to be 0.5. Thus uncertainty is preferred to certain dissimilarity.

Total visual activation The total activation for visual object i is the sum of bottom-up and top-down activations:

$$VA_i = W_{BA} * BA_i + W_{TA} * TA_i$$

W_{BA} and W_{TA} are the weights for the bottom-up and top-down activations respectively. In correspondence with

Wolfe (2007), those weights control the intentional and unintentional attentional captures. Those weights are set to 1.1 and 0.45. The bottom-up activation is given a higher weight to compensate for the distance d_{ij} adjustment, which results in the lower bottom-up activation value in comparison to the top-down activation value.

Saccade and Encoding

After a visual object has been located with a WHERE request, a model can send a WHAT request. This is essentially the same encoding processes of a visual object from the *visicon* as in ACT-R's default vision module. However, PAAV assumes that the saccade that precedes the encoding has a variable execution time dependent on the saccade's amplitude. Prior to a saccade execution, PAAV calculates its duration and landing point. Salvucci (2001) described a set of formulas to calculate those variables. For calculating the execution duration, we used 20 ms as a base execution time and additional 2 ms for an every degree of angular distance between gaze position and the center of the object to be fixated. This is exactly the same method used by Salvucci (2001). Differently from Salvucci (2001), we have used two Gaussian distributions around the center of the object to calculate saccade's landing position. The standard deviation for distribution along X axis is calculated as 0.5 times of the object's linear width. In a similar manner, the standard deviation for Y axis is calculated using object's height. Such implementation is in accordance with theory that the saccade's landing position depends on the size of a visual stimulus (Rayner, 1998).

Upon completion of a saccade, PAAV starts encoding. The encoding time takes a fixed 50 ms. It is in line with findings that the sufficient information is encoded in the first 45-75 ms of a fixation for an object identification to occur (van Diepen, DeGraef, & d'Ydewalle, 1995). Except eccentricity, Salvucci (2001) used word frequency to calculate variable encoding time. However, we believe this approach is not applicable to PAAV where visual object is defined along multiple dimensions. Hence, further study is needed to investigate the object's encoding process in more details sufficient for proper computational modeling.

Visual Decision Threshold

One of the challenging problems in a visual perception is how does the visual system recognize the absence of a desired visual object. For example, humans can spot the absence of a salient object as fast as its presence in a visual field (Figure 1). Similarly, given a WHERE request with specific criteria, how does PAAV know that the desired object is not in iconic memory. One obvious solution is to attend every object in *visicon* and stop when there are no more objects to attend. However, visual search paradigms, such as feature search, show that it is not the case. The visual system is much more efficient and does not require fixation on every item to detect an absence of a target (Treisman & Gelade, 1980; Wolfe, 2007).

PAAV incorporates the concept of a visual decision threshold to decide whether any of the objects in iconic memory will match a given WHERE request. A partial solution is to ignore every object that has zero top-down activation due to complete mismatch. However, results from tasks, such as conjunction search, show that a visual search can be efficient even when distracters partially match the target. PAAV should also be able to filter out objects that match only partially. This is done via simulation of visual grouping based on top-down activation. Given a WHERE request, PAAV returns some object i . Let's assume that, at the time of WHERE request, the distance between object i and the gaze position was d_{Th} , and object i 's top-down activation was TA_{Th} . When object i is encoded these two values are stored and used as a threshold for the consecutive WHERE requests. In the following WHERE requests PAAV completely ignores every object j in iconic memory that has $TA_j \leq TA_{Th}$ and $d_{jg} \leq d_{Th}$ where d_{jg} is a distance between object j and gaze position. Top-down activation serves as a natural threshold for object selection. Every time a model encodes an incorrect object, the acceptance threshold for the next WHERE request increases up to the activation value of that object. The distance d_{Th} provides a measure that PAAV uses to judge whether it can reliably compare two top-down activation values. It is a simulation of a visual grouping where a cluster of similar objects is grouped together. The d_{Th} can be viewed as an approximate radius of the cluster.

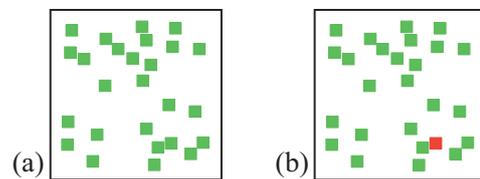


Figure 1: Humans can spot an absence (a) of a red object in field of green objects as fast as its presence (b).

Validation Models

This section describes two models that do common visual tasks. The models are based on ACT-R where the default vision module was replaced with the PAAV module. The tasks are simple, yet demand complex cognitive and perceptual processes, and require most of the components of PAAV module described in this paper. Hence, those tasks serve as a good way to validate the PAAV module.

The first model was created to do feature and conjunction searches. Both of these visual search tasks involve finding a target among a set of distracters. In a feature search task the target differs from distracters by a single feature such as color (Figure 2a). In a conjunction search the target can differ from distracters by either of two features (Figure 2b). A feature search is usually an efficient search with reaction time being independent of a number of distracters. On the other hand, reaction time in a conjunction search increases with a number of distracters. Those results are consistent

among different studies (e.g., Treisman & Gelade, 1980; Wolfe, Cave & Franzel, 1989; Wolfe, 2007).

The second model does a comparative visual search, a paradigm proposed by Pomplun, Sichelschmidt, Wagner, Clermont, Rickheit and Ritter (2001). The task involves detecting a mismatch between two, otherwise equal, halves of a display referred to as hemifields (Figure 3). The task is a simplified version of the traditional picture matching task (Humphrey & Lupker, 1993) with a major difference that it does not require image processing.

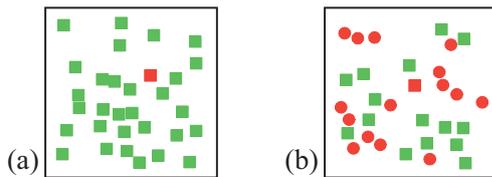


Figure 2: Examples of feature search (a) and conjunction search tasks (b). In both tasks the red rectangle is a target.

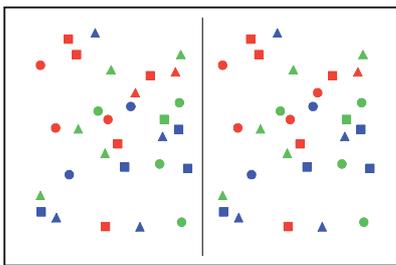


Figure 3: An example comparative visual search task where targets are red triangle and red oval in left and right hemifields respectively.

A Model of Feature and Conjunction Searches

The goal in feature search was to find a red rectangle among green rectangles. In a conjunction search, the model had to find a red rectangle among green rectangles and red ovals. In each trial values for both shape and color were present in near equal amount.

The following experimental conditions were set for the model. In both types of visual search tasks, the set size ranged from 1 to 30. For each set size, there were 500 trials where a target was present and another 500 trials where a target was replaced with a distracter. In total, there were 6000 trials in each of feature and conjunction search tasks. The screen size was $11.3^\circ \times 11.3^\circ$, and the size of each object was 0.85° both in width and height. Within the screen, objects were positioned in a random pattern with the constraint that they should not overlap. The model had to press either “P” or “A” for target being either present or absent. The time of key press was considered as trial end time. The model was reset after each trial.

Figure 4b shows the model’s mean reaction times in both feature and conjunction search tasks each averaged over trials of the same set size. The black solid line is for feature search task where target was present, and black dashed line

is for feature search task where target was absent.

In feature search task the model was asked to find any red object. The resulting RT is mostly independent of set size and averages to 439 ms when a target is present and 640 ms when a target is absent. It is consistent with experimental findings where RT for positive trials is also around 430 ms and for negative trials is 550 ms (Treisman & Gelade, 1980; Wolfe, 2007). The model RT remains the same in positive trials due to very high bottom-up activation the target receives due to its color contrast to homogeneous surrounding objects. Top-down activation from the matching color also contributes to the overall saliency of the target. However, bottom-up activation alone is enough to make the target salient enough to attract almost immediate attention. In negative feature search trials all objects in iconic memory have zero top-down activation. It takes the model few fixations to realize absence of a top-down activation after which the model stops searching. As a result, model also produces flat RT line independent of a set size, although slightly higher than in positive trials.

In a conjunction search task the model was asked to find any red rectangle. Figure 4 compares the RT produced by the model to the RT obtained by Treisman and Gelade (1980) from their experiment with human subjects. As the blue lines in Figure 4 indicate the RT in both positive and negative trials rise as the set size increases. The slopes, however, are different with negative trials having a significantly higher slope. Linear regression of model’s RT on set size gives intercept of 440 ms and 689 ms for positive and negative trials respectively. The slopes are around 19.6 ms/item and 72.8 ms/item. The model results can be compared to those obtained in previous studies (Table 1).

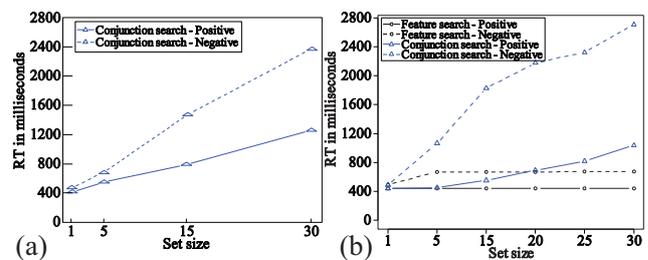


Figure 4: (a) Mean reaction times of human subjects in conjunction search as reported by Treisman and Gelade (1980); (b) Mean reaction times in feature and conjunction search tasks produced by our model.

In this task the distracters are not homogenous. They vary by both color and shape. As a result, there is no guarantee in positive trials that a target will have a higher bottom-up activation than distracters. However, the target always receives higher top-down activation than any other object in iconic memory since it has both matching color and shape. When a set size is small the target’s top-down activation is enough to compensate for smaller bottom-up activation, and the target almost immediately attracts attention as the most salient object. When the set size is big, there is a higher

chance that the target will get significantly lower bottom-up activation than a distracter, which then cannot be compensated by higher top-down activation. Consecutively, those distracters with a higher overall activation are attended first which results in RT increasing with set size.

The main challenge for the model in negative conjunction trials is to know when to stop the search and report the absence of the target. Since most of the distracters either match color or shape with a target, there are few objects that have zero top-down activation. Hence, the model had to rely on visual decision threshold to filter out partially matching distracters. The model requires on average 72.8 ms/item in negative trials indicating that the model does not need to fixate on every object to realize the absence of a target. Hence, top-down activation serves quite well as a visual decision threshold.

Considering the variations between different studies, the model gives a good fit to experimental findings from previous studies with a slightly higher intercept for negative trials than that found in experiments with human subjects. This is probably due to the fact that the corresponding RT line (Figure 4b) is not completely linear, and the elevation for trials with set size of 15 and 20 results in an elevated intercept for an entire linear function. We are still in process of investigating what causes the slightly increased RT for those trials.

Table 1: Comparison of the results of the model's linear regressions of RT on set size to results of linear regression from similar experiments by Treisman and Gelade (1980) and Wolfe, Cave and Franzel (1989).

	Trial type	Slope (ms/item)	Intercept (ms)
Model data	Positive	19.6	440
	Negative	72.8	689
Treisman and Gelade, 1980	Positive	28.7	398
	Negative	67.1	397
Wolfe, Cave and Franzel, 1989	Positive	7.5	451
	Negative	12.6	531

A Model of Comparative Visual Search

For the model of comparative visual search, we set the screen size to $24^\circ \times 16^\circ$, and the size of each object was 0.6° both in width and height. Those are the same conditions used in the original experiment (Pomplun et al., 2001). The screen was divided vertically in two halves, hemifields. Each hemifield contained 30 objects varying in shape (rectangle, oval and triangle) and color (red, green and blue). Each color and shape value was represented in a trial in an equal quantity. Positions of the objects were generated randomly with minimum margin of 10 pixels from the boundaries of the screen. Two hemifields were identical except one object, the target, which mismatched in either color or shape. The target was chosen at random among 30 objects as well as the type of mismatch.

In total, the model had to do 10000 trials where half of the trials had targets that mismatched color and the other half that had targets with mismatched shape. The model was not

aware of the type of mismatch it had to find in a trial. The model was reset after each trial.

The model used a very simple algorithm to do visual search. The model starts from a top-left corner of a screen and does following steps:

1. Fixate on any unattended object (further referred to as O1) in the current hemifield.
2. Fixate on any object (referred as O2) in the opposite hemifield that has the same y coordinate as the O1.
3. If O1 and O2 are the same then go to step 1.
4. If O1 and O2 are different then:
 - a. Fixate on an object NO2 nearest to O2
 - b. Fixate on O1
 - c. Fixate on an object NO1 nearest to O1
 - d. If NO1 and NO2 are the same then end the trial.
 - e. If NO1 and NO2 are not the same then go to step 1.

The steps 4.a to 4.e are necessary to ensure that the module is comparing a correct pair of objects. This uncertainty comes from the fact that when locating a target's twin in the opposite hemifield the model knows only its y coordinate and not the x coordinate. Therefore, it is possible for the model to fixate on a wrong object that by chance had the same y coordinate. To detect such mistakes model also compares two objects from two hemifields that are closest to respective target objects.

The model's mean RT over all trials was 9089 ms (Table 2). On average, the model needed 9007 ms and 9170 ms to finish trials where the difference was either in color or in shape respectively. This is a reasonable fit to reaction times reported by Pomplun et al. (2001). However, the current model was unable to show difference between trials where the mismatch was either in color or in shape.

Table 2: Comparison of model's mean RTs to those reported by Pomplun et al. (2001). All RTs are in ms.

	Color	Shape	Total
Model	9007	9170	9089
Pomplun et al. (2001)	9903	11997	10950

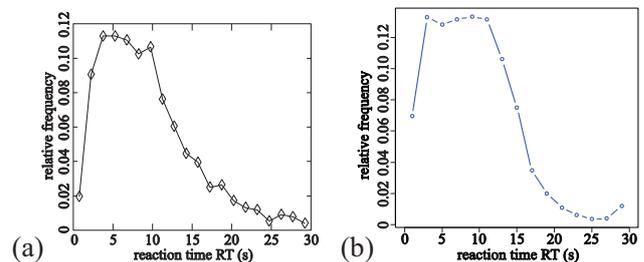


Figure 5: (a) Histogram of reaction times in original comparative visual search experiment (Pomplun et al., 2001); (b) Histogram of reactions times from 10000 model trials in comparative visual search.

Figure 5a shows a histogram of reaction times from original experiment done by Pomplun et al. (2001). This histogram can be compared to a histogram of reaction times

produced by our model depicted in Figure 5b. Both graphs show a plateau of short reaction times between three and ten seconds, indicating that the distribution of RT produced by the model closely fits the distribution from the original experiment. On average, the model made 37.3 fixations during a trial. This is a close match to 39.6 fixations reported by Pomplun et al. (2001). The model produces nicely structured scanpath (Figure 6) even though there is no explicit control of which object should be chosen as O1.

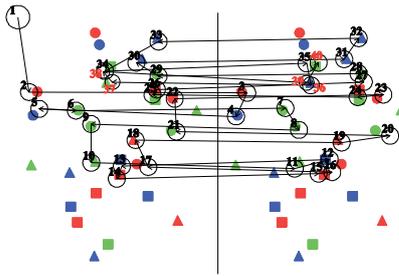


Figure 6: Example scanpath produced by the model. Open circles indicate fixations while arrows indicate saccade directions. Numbers are positions of fixations in the fixation sequence. Targets are blue and green triangles at 36th and 37th fixations.

Conclusion

There are many existing models of the human visual system. We have greatly leveraged from those models by adopting different concepts and integrating them into one module that became PAAV. Our main goal is not to reinvent the wheel, but to create a tool that allows modelers to create cognitively plausible models of tasks that require comprehensive visual system. This is the major difference between PAAV and existing models of a visual system. Models, such as a three-level model of comparative visual search (Pomplun & Ritter, 1999) or Guided Search 4.0 (Wolfe, 2007), were created to perform very specific set of tasks. On the other hand, PAAV was developed to be general enough to model a wide range of tasks. This is why we prefer to call PAAV a module rather than a model. Furthermore, PAAV is not a stand-alone tool, but rather a part of a cognitive architecture. For example, Guided Search 4.0 excels at modeling feature and conjunction search tasks. However, an absence of a general cognitive theory makes it hard to investigate top-down influence in these tasks. On the other hand, ACT-R imposes limitations on what PAAV is allowed to do, but it also gives additional layer of plausibility. The source code for the PAAV module and the models of the visual search tasks described in this paper can be downloaded via <http://ai178174.ai.rug.nl/iccm2012/>.

References

Anderson, J. R. (2007). *How Can Human Mind Occur in the Physical Universe?* New York: Oxford University Press.

- Clark, A. (1997). *Being There: Putting Brain, Body and World Together Again*. Cambridge, MA: MIT Press.
- Humphrey, G. K., & Lupker, S. J. (1993). Codes and Operations in Picture Matching. *Psychological Research*, *55*, 237-247.
- Kieras, D. (2010). Modeling Visual Search of Displays of Many Objects: The Role of Differential Acuity and Fixation Memory. *Proceedings of the 10th International Conference on Cognitive Modeling*, (pp. 127-132).
- Kieras, D. (2009). The Persistent Visual Store as the Locus of Fixation Memory in Visual Search Tasks. In A. Howes, D. Peebles, & R. Cooper (Ed.), *9th International Conference on Cognitive Modeling*. Manchester, UK.
- Land, M., Mennie, N., & Rusted, J. (1999). The Roles of Vision and Movements in the Control of Activities of Daily Living. *Perception*, *28*, 1311-1328.
- Nyamsuren, E., & Taatgen, N. A. (2011). Top-down Planning and Bottom-up Perception in a Problem-solving Task. *Proceedings of the 33rd Annual Conference of the Cognitive Science Society* (pp. 2685-2690). Austin, TX: Cognitive Science Society.
- Pomplun, M., & Ritter, H. (1999). A Three-level Model of Comparative Visual Search. *Proceedings of the Twenty First Annual Conference of the Cognitive Science Society*, (pp. 543-548).
- Pomplun, M., Sichelschmidt, L., Wagner, K., Clermont, T., Rickheit, G., & Ritter, H. (2001). Comparative Visual Search: A Difference That Makes a Difference. *Cognitive Science*, *25* (1), 3-36.
- Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, *124*, 372-422.
- Salvucci, D. D. (2001). An Integrated Model of Eye Movements and Visual Encoding. *Cognitive Systems Research*, *1* (4), 201-220.
- Theeuwes, J. (1992). Perceptual selectivity for color and form. *Perception & Psychophysics*, *51*, 599-606.
- Treisman, A., & Gelade, G. (1980). A Feature-integration Theory of Attention. *Cognitive Psychology*, *12*, 97-136.
- van Diepen, P. M., De Graef, P., & d'Ydewalle, G. (1995). Chronometry of foveal and peripheral information encoding during scene perception. In J. M. Findlay, R. Walker, & R. W. Kentridge (Eds.), *Eye movement research: Mechanisms, processes and applications* (pp. 349-362). Amsterdam: North Holland.
- Wolfe, J. M. (2007). Guided Search 4.0: Current Progress With a Model of Visual Search. In W. Gray (Ed.), *Integrated Models of Cognitive Systems* (pp. 99-119). New York: Oxford.
- Wolfe, J. M., Cave, K. R., & Franzel, S. L. (1989). Guided Search: An Alternative to the Feature Integration Model for Visual Search. *Journal of Experimental Psychology: Human Perception and Performance*, *15*, 419-433.
- Wolfe, J., & Horowitz, T. (2004). What Attributes Guide the Deployment of Visual Attention and How Do They Do It? *Nature Reviews Neuroscience*, *5* (6), 495-501.

ACTR-QN: Integrating Queueing Network and ACT-R Cognitive Architectures

Shi Cao (shicao@umich.edu) & Yili Liu (yililiu@umich.edu)

University of Michigan
Ann Arbor, Michigan 48109 USA

Abstract

Integrating the symbolic modeling capabilities of ACT-R and the mathematical and visualization capabilities of Queueing Network has both theoretical and applied values. Theoretically, the integrated ACTR-QN allows modelers to examine a wider range of fundamental cognitive issues from new perspectives. For cognitive engineering applications, a software program implementing ACTR-QN has been developed, including task templates, model setup assistants, and visualization functionalities. These tools and features support easy model building and intuitive model analysis of both task performance and mental workload.

Relating ACT-R buffer activation to EEG activity during an attentional blink task

Marieke K van Vugt (m.k.van.vugt@rug.nl)

ALICE, Nijenborgh 9

9747 AG Groningen

The Netherlands

Abstract

While a clear relation has been established between ACT-R and activity in fMRI, little is known about whether ACT-R has also correlates in EEG activity. Because of its superior temporal resolution compared to fMRI, EEG could potentially be used to adjudicate between model versions that differ in time courses of module activation, even while generating qualitatively similar patterns of behavioural data. On the other hand, ACT-R could form a much-needed source for hypotheses about interactions between brain areas (synchronization) in EEG data. I discuss a method to find such a mapping between ACT-R and EEG buffers, and apply it to data from an attentional blink experiment (Martens et al., 2006). I show preliminary EEG correlates of ACT-R modules and discuss broader implications of this approach for both cognitive neuroscience and cognitive modeling with ACT-R.

Keywords: EEG; ACT-R; attentional blink; oscillations

Introduction

There is a growing interest in using neural activity to help in constraining cognitive models and for cognitive models to help understand the brain. One of the models for which this brain-to-model mapping has worked very well is the ACT-R cognitive architecture (Anderson, Fincham, Qin, & Stocco, 2008). Multiple experiments have verified this mapping, and conversely, the mapping of ACT-R to fMRI (functional magnetic resonance imaging) has given rise to interesting neural predictions.

Despite the success of the mapping between ACT-R and fMRI, there has not been a comparable mapping between ACT-R and EEG (electroencephalography) data. EEG differs from fMRI in that it has a much higher temporal resolution (on the order of milliseconds) compared to the supra-second resolution of fMRI. This increase in temporal resolution of EEG compared to fMRI is countered by a decrease in spatial resolution. While fMRI is very well-suited to answer questions about what parts of the brain are associated with the different ACT-R modules, EEG could answer questions about differences in their time courses of activation. This is interesting because candidate cognitive models could differ in the time course of activation of various buffers, but this may not yield to observable differences in behaviour. An example of this concerns the question of how long the retrieval buffer takes to turn off after activation. Varying the retrieval buffer's decay time does not lead to qualitatively different predictions for behaviour in most experiments. Nevertheless, these models could potentially still be distinguished with a tool like EEG, which has a very high temporal resolution.

While EEG is most conventionally analyzed in terms of event-related potentials, i.e., the average electric field measured in an electrode in response to a certain event, another

way is to examine electrical activity in different frequency bands that need to necessarily be time-locked to an event. It has been proposed that such oscillatory activity can be used to communicate and bind information across different parts of the brain (e.g., Singer, 1993). To have a more comprehensive grasp of EEG activity, we will consider both oscillatory and non-oscillatory EEG in our work.

Although a no mapping has been made between EEG activity and all ACT-R modules, some authors have proposed electrophysiological correlates for the production system that forms the core of ACT-R. For example, Zylberberg, Dehaene, Roelfsema, and Sigman (2011) propose that the ACT-R production system is similar to the Global Neural Workspace hypothesis in that the cognitive system selects productions serially from a set of sensory, memory, and motor options. Selection is mediated by mutual inhibition between neurons that increase in activation until a threshold is reached. Notably, "production selection resembles single decision making" (Zylberberg et al., 2011). A lot is known about the neural correlates of making a single decision between multiple alternatives which provides hypotheses for the neural correlates of production selection ("deciding" between productions). I have previously proposed that evidence accumulation is associated with power of oscillatory activity in the 4–9 Hz theta band in EEG (van Vugt, Simen, & Cohen, 2011) and crossing a threshold with the Lateralized Readiness Potential (an EEG potential consisting of the imbalance between the left and right-hemisphere central electrodes C3 and C4 that is thought to arise from motor cortex, see Figure 1; Simen, van Vugt, Balci, Freestone, & Polk, 2010; van Vugt, Simen, Nystrom, Holmes, & Cohen, submitted). Simen et al. (2010) also proposed that production selection would be associated with the Lateralized Readiness Potential.

In this study, we look for the electrophysiological correlates of a larger set of ACT-R modules in an attentional blink task, for which a well-established ACT-R model exists (Taatgen, Juvina, Schipper, Borst, & Martens, 2009). In an attentional blink task (Luck, Vogel, & Shapiro, 1996) participants see a very rapid stream of visual stimuli, and have to detect what letters were presented in this stream of digits. The main finding of interest in this task is that while participants can see two letters if they occur far apart or in direct succession, they often fail to see the second letter if it is separated from the first by one or two intervening digits. It is as if attention blinks after seeing the first letter. ACT-R accounts for the attentional blink phenomenon by assuming there is an overexertion of control. If, when a target is recognized in the stream of stimuli, a control rule is triggered in the production

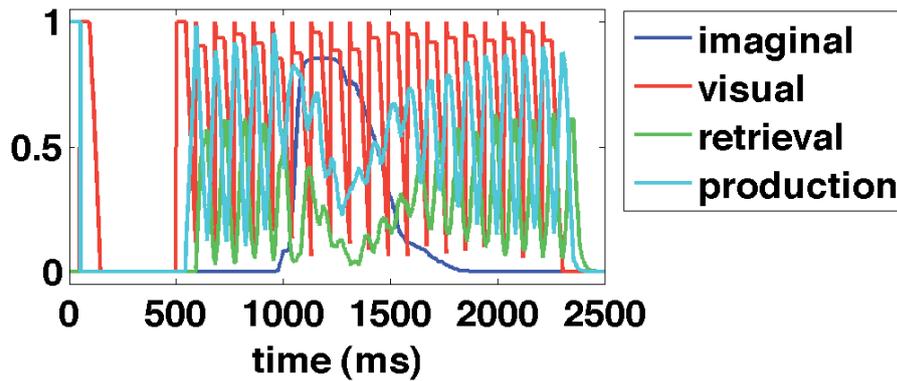


Figure 1: Example module activation probabilities for the ACT-R model of the attentional blink (lag 3, correct trials). Blue: imaginal module. Red: visual module. Green: retrieval module. Cyan: production module. These module activations were used to create the ACT-R regressors that were correlated with the EEG data.

module that suspends target detection, then this can create an attentional blink because the imaginal buffer is not open for receiving another target to consolidate during the “suspend target detection” time.

There have been two main findings in EEG studies of the attentional blink: an increase in the P3 event-related component (the P3 is a positive potential occurring approximately 500 ms after a stimulus onset at parietal electrode sites), and a decrease in gamma oscillation synchronization. The increase in the P3 has also been associated with an increase in 4–9 Hz theta oscillation reset, and has been thought to reflect over-investment of attentional resources in the first target stimulus (Slagter et al., 2007). This phenomenon may be similar to the over-exertion of control posited by the ACT-R model, and may be associated with the imaginal module. The decrease in gamma synchronization was predicted by the Global Neural Workspace model by Dehaene, Sergent, and Changeux (2003), which as discussed above, shares conceptual commonalities with ACT-R. Gamma oscillations are periodic activity observable in the EEG at a frequency of 28–90 Hz. Gamma oscillations have been associated with many things, including visual attention and consciousness (Varela, Lachaux, Rodriguez, & Martinerie, 2001). According to Dehaene’s model, when gamma synchronization decreases, it makes the visual stimulus less accessible to consciousness (Gaillard et al., 2009), and hence the participant will frequently fail to report that s/he has seen the stimulus. In terms of ACT-R, this may reflect an inability of the visual stimuli to enter the imaginal buffer.

My goal is to examine whether we can find neural correlates of ACT-R during the attentional blink in EEG data. Guided by the above observations, I predict that activation of the imaginal module, which is crucial for the attentional blink effect, is correlated with 4–9 Hz theta oscillations and the P3 EEG component. I further predict that the gamma synchronization decrease that is also associated with the attentional blink reflects a disconnection between the visual module and

the retrieval module, such that items entering the visual module cannot be compared to memory (chunk activation from items in the visual buffer cannot spread to chunks in declarative memory during a retrieval request). Nevertheless, in testing these hypotheses, I will look at all frequency bands because there exist other plausible hypotheses and the field is relatively unexplored.

Methods

Task: I used existing data from an attentional blink task (Martens, Munneke, Smid, & Johnson, 2006) to study the electrophysiological correlates of ACT-R. In this task, participants see a very rapid stream of visual stimuli, presented for 90 ms each. Their task is to report whether there are letters present in the stream, and if so, which letters those are. The data reported here are from the 14 blinkers in the study by Martens et al. These EEG data were collected at the University of Groningen with a 64-channel EEG system (Twente Medical Systems, Enschede, The Netherlands) and a sample rate of 250 Hz.

Analysis: EEG data were analyzed with the EEG toolbox, a set of Matlab scripts developed in the laboratory of Michael Kahana (e.g. van Vugt, Schulze-Bonhage, Litt, Brandt, & Kahana, 2010) and custom-written scripts. I used this toolbox to extract data for every channel in our EEG setup. I concatenated the time series for each trial lengthwise into one long time series to be correlated with the ACT-R model time series. I then used Morlet wavelets (van Vugt, Sederberg, & Kahana, 2007) to create representations of the EEG data in six distinct frequency bands: 2–4 Hz delta, 4–9 Hz theta, 9–14 Hz alpha, 14–28 Hz beta, 28–48 Hz low gamma and 48–90 Hz high gamma (van Vugt et al., 2010). For this frequency-transformed data, I used the same concatenation procedure to create time series of the trial EEG for each standard frequency band.

To correlate ACT-R’s predicted module dynamics to EEG data, I created regressors (van Vugt et al., 2011). Regressors

Table 1: Predictions for the neural correlates of ACT-R modules based on the cognitive neuroscience literature. Note that the speech module would create large artifacts in EEG activity, making it difficult to find correlates for this buffer. Reported fMRI correlates are based on Anderson et al (2008) and Borst et al (2011).

¹ Keeping track of subgoals and intentions. ² ACC = Anterior Cingulate Cortex. ³ Storing and retrieving declarative information. ⁴ Auditory perception.

Module	fMRI region	EEG component
Motor module	Motor cortex	Central beta oscillations
Vision module	Fusiform gyrus	Posterior gamma oscillations
Imaginal module ¹	Parietal cortex (Intraparietal Sulcus)	Parietal theta oscillations
Goal module	ACC ²	Frontal theta/gamma oscillations
Retrieval module ³	Lateral inferior prefrontal	Hippocampal theta oscillations
Speech module	Artifacts	Artifacts
Aural module ⁴	Secondary auditory cortex	Central gamma oscillations
Production selection	Head of caudate	Lateralized Readiness Potential

are fMRI terminology for a time series of interest that is used as the independent variable in a regression to find pieces of neural data that correspond to these dynamics. In this case, the data patterns of interest are ACT-R module activations (visual, production, retrieval, and imaginal). I ran the attentional blink ACT-R model (Taatgen et al., 2009) 250-350 times (corresponding to the number of trials in the dataset) and computed the average activation for different model conditions: lag 3 and 8, and correct and incorrect responses. These average activations therefore reflect the probability of a module being active. 'Lag' refers to the number of stimuli between the first and second target (letter) in the digit stream that the participant has to remember. An attentional blink is likely to occur for lag 3, but not lag 8 trials. Correct trials refer to trials in which both targets were reported correctly. Trials in which the first target was missed were removed from the analysis because in that case it is not clear what the reason is for missing the second target if that occurs.

For every trial that a participant did, I inserted the averaged module activation for the condition corresponding to that trial. This led to an activation time series during the whole tasks for every ACT-R module that, after subsampling to the EEG sample rate (250 Hz), had the same length as the EEG data. These were the time series that I could use to regress the EEG time series on, to obtain for every module an estimate of how well it correlated with the different frequency bands, and which channels were most strongly involved in this correlation. Instead of using a simple multiple regression, I used a canonical correlation analysis, which finds weights on the regressors (electrode time series) that maximize the correlation between the regressors and the ACT-R time series. Note that only the EEG data were wavelet-transformed—the ACT-R activation time courses were not.

Predictions: I correlated the regressors representing ACT-R's module dynamics not only with raw EEG, but also with oscillatory EEG. Oscillatory activity has two advantages over event-related raw EEG activity: the mechanisms that produce field potential oscillations are well-understood, and because many phenomena get lost in averaging, time-frequency de-

Table 2: Cross-correlations for the various ACT-R module time courses, averaged across participants. The activation time course of the imaginal module is most different from the other modules, as would be predicted from the module time courses in Figure 1.

	Imaginal	Production	Retrieval	Visual
Imaginal	1.0000	0.3922	0.3057	0.3880
Production	0.3922	1.0000	0.2138	0.5046
Retrieval	0.3057	0.2138	1.0000	0.5238
Visual	0.3880	0.5046	0.5238	1.0000

composition allows for a more comprehensive and in-depth picture of the data (Cohen, Wilmes, & van de Vijver, 2011). It is well possible that different modules are associated with different oscillatory frequency bands. Table 1 shows my hypotheses about correlations between ACT-R modules and components of EEG activity based on the EEG literature.

Results

The basic behavioural and EEG data for this task are reported in Martens et al. (2006), who showed a classic attentional blink effect (dip in accuracy for the second target letter when it followed the first target letter with only 1 or 2 items in-between). This was accompanied by an increased P3 EEG component for blinked compared to non-blinked trials.

Figure 1 shows an example of average ACT-R module activation on a single trial for the lag 3/correct condition. These activation time courses were used to make regressors that could be used to extract corresponding patterns from our EEG data. I correlated these regressors with both the raw EEG data and oscillatory data in the different frequency bands. Figure 2 shows the resulting correlations between each module and EEG activity for all participants who showed evidence of an attentional blink in the task. Note that the activations of the visual, retrieval and production modules are highly correlated (Table 2; see also Borst, Taatgen, and Rijn (2011, for a discussion)) and are therefore expected to have very similar neural correlates. The highest correlations occur with activity

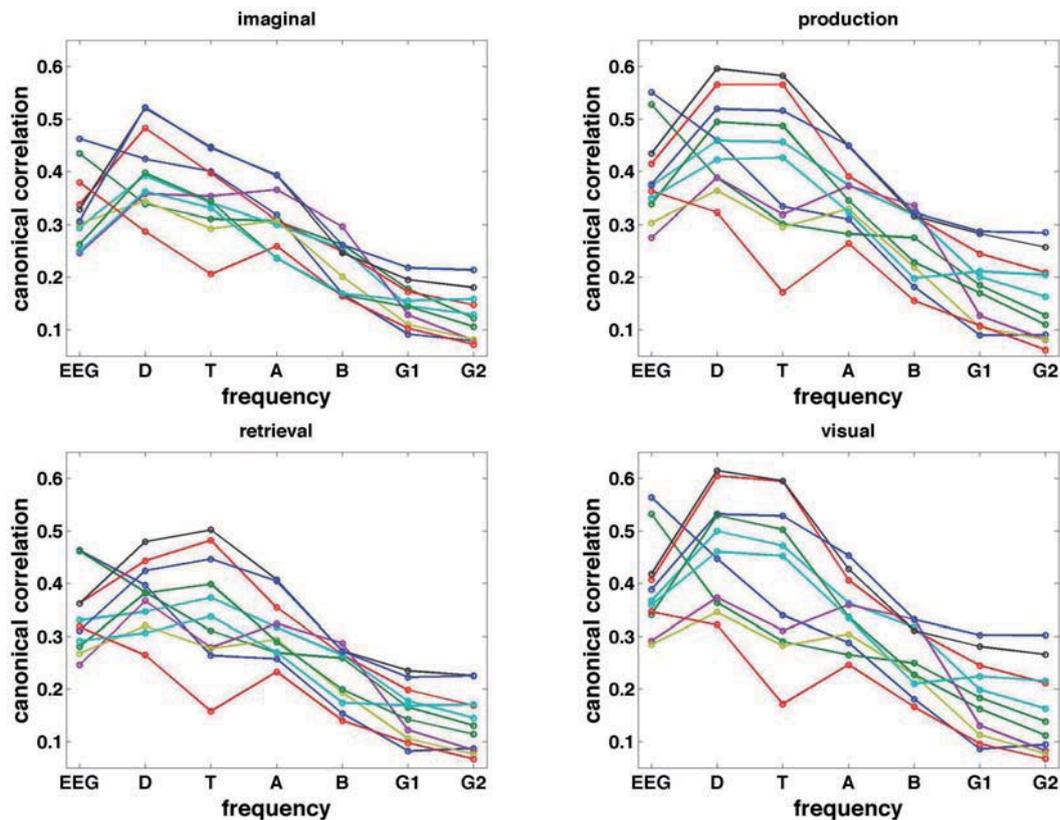


Figure 2: Canonical correlation between ACT-R module time courses and EEG time courses for raw EEG (“EEG”), 2–4 Hz delta (“D”), 4–9 Hz theta (“T”), 9–14 Hz alpha (“A”), 14–28 Hz beta (“B”), 28–48 Hz low gamma (“G1”) and 48–90 Hz high gamma (“G2”) activity. Different lines reflect different participants. As expected from the correlations between module activations, the imaginal module shows a different pattern (most prominent correlation with EEG activity in the delta band) from the other modules (most prominent correlation with EEG activity in the theta band).

in the 4–9 Hz theta band for the visual, retrieval, and production modules. For the imaginal module, we additionally see a fairly high correlation with activity in the 2–4 Hz theta band as well. Overall canonical correlations are lowest for the retrieval module. Interestingly, there are three participants for whom raw EEG activity shows the highest canonical correlation, while for all others oscillatory EEG shows the highest canonical correlation. In contrast to our expectations, the different modules in this experiment do not exhibit correlations with distinct frequency bands. Part of this may be due to the relatively high correlations between module activation time courses.

I then examined the topographies associated with the magnitudes of ACT-R–EEG correlations in the different frequency bands. While the correlations look quite similar across modules and frequency bands, the topographies in Figure 3 show more variation. In this graph, I chose for each module a frequency band based on either the magnitude of the correlation of the EEG with the module activation in Figure 2 or based on the hypotheses in Table 1. I found that the imaginal module correlate in the 4–9 Hz theta band was primarily associated with right-lateral activation that could be

consistent with a parietal source as I expected. The production module correlate in the 4–9 Hz theta band was found predominantly in superior channels that are in the same location as where Lateralized Readiness Potentials are observed. The retrieval module correlate in the 2–4 Hz delta band showed a negative correlation in frontal channels, consistent with a correlate of the retrieval module in frontal cortex (but unlike my prediction of hippocampal theta oscillations, although those are virtually impossible to observe on the scalp). It also showed a positive correlation with right-lateral channels. Finally, the visual module correlate in the 28–48 Hz gamma band had a central topography, which is quite different from the occipital locus I expected for this module. The gamma band correlate was also much weaker than the correlation in the delta and theta bands, which may therefore be much more likely correlates of this buffer.

Discussion

I have proposed a new method to find the electrophysiological correlates of ACT-R module activations, and shown that different buffers show different patterns of correlation with EEG data. Not all the predictions in Table 1 have been ver-

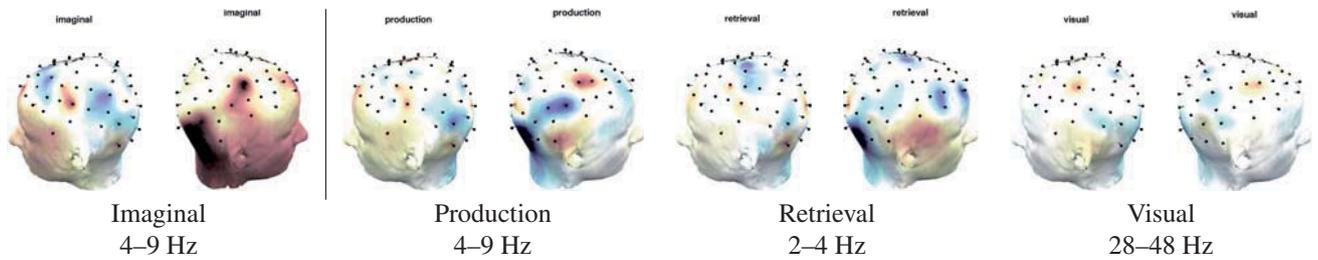


Figure 3: Topographical representation of electrode weights from the canonical correlation analysis for the different modules. The choice of frequency band was guided by the canonical correlation observed in Figure 2. Plotted are the magnitudes of the canonical correlation weights across the brain for the canonical correlation at the respective frequency and with the respective module. Positive weights are red and negative weights are blue.

ified. The data reported here support the idea that the imaginal module is associated with parietal theta oscillations, but do not support the mappings for visual and retrieval modules. The visual module seems to instead be correlated more strongly with frontal gamma oscillations, just like the retrieval module. Interestingly, this mapping of the retrieval module is fairly consistent with fMRI-based localization. The association of the production module with central theta oscillations could be consistent with an association with Lateralized Readiness Potentials, but this should be tested more explicitly. What further remains to be done is applying the same methods to different tasks and to optimize artifact detection methods. In particular, it is important to look at tasks in which the production, retrieval and visual module are not as highly correlated as they are in the attentional blink task discussed here, such that their neural correlates can be pulled apart. Only this will allow us to make claims about the electrophysiological correlates of ACT-R's various resources.

Areas that warrant further investigation are modeling individual differences and examination of the neural correlates of module interaction. Individual differences could solidify our confidence in the mapping between modules and EEG activity. If individual differences are modeled in ACT-R (e.g., Lovett, Daily, & Reder, 2000) and if those individual differences correlate with individual differences in those participants' electrophysiology, then this makes the EEG–ACT-R-module relation more specific (see van Vugt et al., 2011, for an application of this approach to perceptual decision making). In other words, if individual differences covary with dynamics of the neural ACT-R modules, that could greatly increase our confidence in the accuracy of our mapping of ACT-R to electrophysiological brain activity.

Once the electrophysiological correlates of ACT-R have been determined, a large area of new research is opened up. An advantage of the fact that ACT-R consists of multiple modules is that their interaction provides a principled way to look for patterns of synchronization in EEG activity. If synchronization reflects information transfer between the modules (Buzsáki, 2006; Singer, 1993), then increases of synchronization should occur in specific frequency bands and between specific sets of electrodes that correspond to the re-

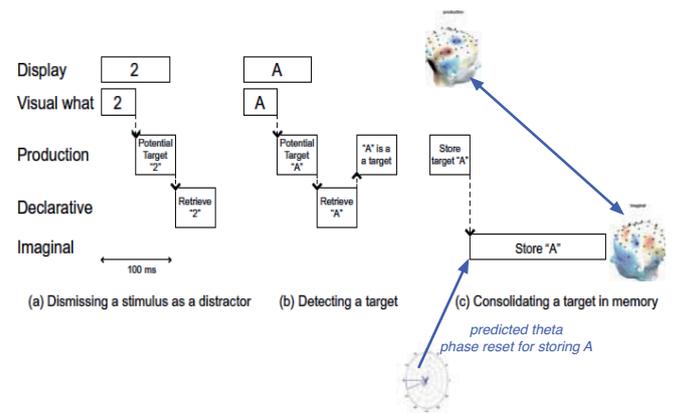


Figure 4: Predictions for neural correlates of interaction between modules in the Attentional Blink task (adapted from Taatgen et al., 2009). At the time of storing the letter “A” in the imaginal buffer, there should be phase reset (observable in the phase histogram) and subsequent synchronization in the theta/alpha band between the channels corresponding to the production and imaginal modules. These sets of channels were based on Figure 3.

spective modules. For the modules in Figure 4, for example, I predict that after every stimulus presentation, there should first be increased synchronization between the neural correlates of the visual module and the production module, and then between the production and declarative module (panel a). When a target is stored successfully, but not when it is not, there should be increased synchronization between the production and the imaginal module (panel c).

I believe that relating ACT-R to EEG activity is a fruitful endeavor that could eventually also have interesting implications for ACT-R modeling. For example, there might be subtle differences in module activation that may not lead to observable differences in behavior. If we could observe module activation time courses in EEG, this could potentially al-

low us to distinguish between these different ACT-R models. While fMRI has had tremendous success in defining brain regions associated with different ACT-R modules, it does not have the temporal resolution on millisecond-scale to compare different time courses of module activation. EEG could fill this gap. Moreover, EEG is much better suited to capture brief interactions between ACT-R modules, which would be too short for fMRI to detect.

In conclusion, I have outlined methods to study the neural correlates of ACT-R in electrophysiological data. I have also shown how they work in the case of an attentional blink task, and how different ACT-R modules can be associated with specific frequency bands and topographies observable in EEG data. I argue that these methods can lead to a wealth of understanding on how the time courses of ACT-R models develop over time. Moreover, they could provide neuroscientists with directly-testable hypotheses about interaction between different neural populations.

Acknowledgments

I gratefully acknowledge Patrick Simen for providing useful discussion and seminal ideas for this work. I also thank all members of the cognitive modeling group, in particular Niels Taatgen, Stefan Wierda, Ronald van Elburg, Jurjen Wierda, Hielke Prins, and Hedderik van Rijn, for useful discussion on this project.

References

- Anderson, J. R., Fincham, J. M., Qin, Y., & Stocco, A. (2008). A central circuit of the mind. *Trends in Cognitive Sciences*, 12(4), 136–143.
- Borst, J., Taatgen, N., & Rijn, H. van. (2011). Using a symbolic process model as input for model-based fmri analysis: Locating the neural correlates of problem state replacements. *NeuroImage*, 58(1), 137–147.
- Buzsáki, G. (2006). *Rhythms of the brain*. New York: Oxford University Press.
- Cohen, M. X., Wilmes, K. A., & van de Vijver, I. (2011). Cortical electrophysiological network dynamics of feedback learning. *Trends in Cognitive Sciences*, 15(12), 585–566.
- Dehaene, S., Sergent, C., & Changeux, J.-P. (2003). A neuronal network model linking subjective reports and objective physiological data during conscious perception. *Proc. Nat. Acad. Sci., USA*, 100(14), 8520–8525.
- Gaillard, R., Dehaene, S., Adam, C., Clémenceau, S., Hasboun, D., Baulac, M., et al. (2009). Converging intracranial markers of conscious access. *PLoS Biology*, 7(3), e1000061.
- Lovett, M. C., Daily, L. Z., & Reder, L. M. (2000). A source activation theory of working memory: cross-task prediction of performance in ACT-R. *Cognitive Systems Research*, 1(2), 99–118.
- Luck, S. J., Vogel, E. K., & Shapiro, K. L. (1996). Word meanings can be accessed but not reported during the attentional blink. *Nature*, 383, 616–618.
- Martens, S., Munneke, J., Smid, H., & Johnson, A. (2006). Quick minds don't blink: Electrophysiological correlates of individual differences in attentional selection. *Journal of Cognitive Neuroscience*, 18(9), 1423–1438.
- Simen, P., van Vugt, M. K., Balci, F., Freestone, D., & Polk, T. (2010). Toward an analog neural substrate for production systems. In *Proceedings of the international conference in cognitive modeling*.
- Singer, W. (1993, March). Synchronization of cortical activity and its putative role in information processing and learning. *Annu Rev Physiol*, 55, 349–374.
- Slagter, H. A., Lutz, A., Greischar, L. L., Francis, A. D., Nieuwenhuis, S., Davis, J. M., et al. (2007). Mental training affects distribution of limited brain resources. *PLoS Biology*, 5(6), e138.
- Taatgen, N. A., Juvina, I., Schipper, M., Borst, J. P., & Martens, S. (2009). Too much control can hurt: A threaded cognition model of the attentional blink. *Cognitive Psychology*, 59(1), 1–29.
- van Vugt, M. K., Schulze-Bonhage, A., Litt, B., Brandt, A., & Kahana, M. J. (2010). Hippocampal gamma oscillations increase with working memory load. *Journal of Neuroscience*, 30, 2694–2699.
- van Vugt, M. K., Sederberg, P. B., & Kahana, M. J. (2007). Comparison of spectral analysis methods for characterizing brain oscillations. *Journal of Neuroscience Methods*, 162(1–2), 49–63.
- van Vugt, M. K., Simen, P., & Cohen, J. D. (2011). Finding neural correlates of drift diffusion processes in eeg oscillations. In L. Carlson, C. Hölscher, & T. Shipley (Eds.), *Proceedings of the 33rd annual conference of the cognitive science society* (p. 2439–2444). Austin, TX: Cognitive Science Society.
- van Vugt, M. K., Simen, P., Nystrom, L. E., Holmes, P., & Cohen, J. D. (submitted). Lateralized readiness potentials reveal properties of a neural mechanism for crossing a decision threshold. *submitted*.
- Varela, F., Lachaux, J. P., Rodriguez, E., & Martinerie, J. (2001). The brainweb: Phase synchronization and large-scale integration. *Nature Reviews Neuroscience*, 2(4), 229–239.
- Zylberberg, A., Dehaene, S., Roelfsema, P. R., & Sigman, M. (2011). The human Turing machine: a neural framework for mental programs. *Trends in Cognitive Sciences*, 15(7), 293–300.

One Model, Two Tasks: Decomposing the Change Signal Task

L. Richard Moore Jr. (larry.moore@L-3com.com)

L3 Communications
5950 East Sossaman Road, Suite 102
Mesa, AZ 85212 USA

Glenn Gunzelmann (glenn.gunzelmann@us.af.mil)

Cognitive Models and Agents Branch, Air Force Research Laboratory
Wright Patterson Air Force Base, OH 45434 USA

Marissa Daigle (mdaigle@g.clemson.edu)

Clemson University
Department of Mathematical Sciences
Clemson, SC 29634 USA

Abstract

The change signal task is a variant of a two-alternative forced-choice (2AFC) task where the initial stimulus is superseded with the alternative stimulus (the change signal) at a delay on a proportion of trials. Taking advantage of the overlap in task requirements, we present a single model that can perform both tasks. We validate the model using the empirical data from participants who performed both tasks sequentially. The results confirmed the existence of a dynamic hedging strategy, and showed that cognitive fatigue had little, if any, role in slower response times with increased time on task. When fitting the 2AFC task, the model required adjustment to one architectural parameter while the rest were left to defaults. That parameter is then constrained while fitting the remaining three task-specific parameters of the change signal task. This effectively reduces a degree of freedom in the larger task, and increases confidence in the model as it closely matches human performance in multiple tasks.

Keywords: ACT-R; change signal; two-alternative forced-choice; cognitive model.

Introduction

Roberts and Pashler (2000) point out that modelers should consider criteria other than fit when evaluating the credibility of a model. Specifically, they propose examining the data that the model is unable to fit: how much data disagree with the theory, and how strongly does it do so? And could the model fit any data?

Within the computational cognitive modeling community, other approaches to bolster model confidence exist. For example, an established cognitive architecture (Anderson, 2007; Rosenbloom, Laird & Newell, 1993; Meyer & Kieras, 1997) provides the software framework to constrain a model to specific theories of cognitive processes that have been validated independently in the literature. With an active cognitive architecture user community, models can be tested against new empirical data and different experimental conditions (Gobet & Ritter, 2000; Gunzelmann, Byrne, Gluck & Moore, 2009; Gunzelmann, Moore, Salvucci & Gluck, 2011). Even when parameters are adjusted to account for individual and group differences, successful

model reuse instills confidence in the theory behind the model.

More complex models provide another validation opportunity through task decomposition. For example, Myers (2009) reports on a composite model that integrates two previously published models to perform a more complex task. Independent validation of each sub-task instills greater confidence in the composite model (cf., Halverson & Gunzelmann, 2011).

In this paper, we present a single model that can perform two tasks with the same set of knowledge. Model performance on the simpler task relies entirely on a subset of knowledge from the more complex task. Because of this relationship, the simpler model can be fit to empirical data, and we would expect any relevant architectural parameters to be identical for the complex task when fitting performance of the same participants on both tasks. Even though the complex task introduces several task-specific parameters that require fitting, all architectural parameters are constrained by the simpler task fit.

The proposed fitting strategy requires a suitably designed empirical study. Specifically, a repeated measures design where each participant performs both tasks in sequence is a necessity. Therefore, we also report on a study that allows for the independent task fitting approach described above.

Change Signal Task

The *change signal task* (Brown & Braver, 2005) and the *two-alternative forced choice* (2AFC) task provide the context for the empirical study as well as the model discussed in this paper. The 2AFC task is the simpler of the two, yet it provides all the necessary fundamentals to perform the more complex change signal task. (There is, however, a parameterized difference in strategy between the two tasks that is discussed in the model fitting section below.)

The change signal task was originally devised by Brown and Braver (2005) as a variation of the classic Logan and Cowan (1984) *stop signal task*. Whereas the stop signal task focused on response inhibition, the Brown and Braver variant focused on changing responses.

At its core, the change signal task is a modified 2AFC task. In the basic 2AFC design, participants respond to arrows pointing right or left by pressing the associated arrow key on the keyboard. The modification for the change signal task is that on 1/3 of the trials, a larger arrow appears after the initial stimulus, critically timed to interrupt their normal response. The larger arrow always points in the opposite direction of their initial response, and signals participants to inhibit their initial response and instead respond to the change signal.

In Brown and Braver (2005), the timing of the change signal was dynamically adjusted to induce consistent error rates. In fact, the task implemented two change stimulus delays to produce different error rates: a 50% high error rate condition and a 4% low error rate condition. These two conditions were not explained to the participants prior to the experiment, but they were differentiated by stimulus colors (color cue conditions).

The experimental tasks are described in more detail below, as well as the modeling insights gleaned.

Empirical Work

In Moore, Gunzelmann, and Brown (2010), we examined the Brown and Braver (2005) data and found that participants responded more slowly over time. We proposed that subjects were “hedging” their responses in anticipation of a possible change signal, and that participants hedged for longer periods of time with increased task experience. We also raised the possibility that some slowing may be the result of time on task effects, but the data from the change signal task alone did not allow us to assess that possibility in detail. One motivation for the study described in this paper was to evaluate the role of within task fatigue more thoroughly.

Another important result in the Brown and Braver study was that participants responded to the high error rate condition more slowly (allowing more time for the change signal to appear) compared to the low error rate condition. One interpretation of these data is that participants were forming an implicit association of stimulus color to error condition, which was implemented in our earlier model (Moore et al., 2010). However, in this paper we will demonstrate that this is not necessarily the case, and present a model that embodies a more parsimonious explanation for the data. This will be discussed in the results section.

Experiment

The experiment included 33 participants between the ages of 18 and 50, with 18 females and 15 males. Participants were asked to perform two tasks during the hour-long experiment. One was the change signal task, and the second was a 2AFC task. The order of the two tasks was counterbalanced across the participants. All participants completed 642 trials for each task, except one who mistakenly only completed the change signal task. Data from that subject is excluded from this paper.

At the start of the experiment, participants were shown instructions and allowed to perform six sample trials for each task. Instructions for each task were redisplayed before the participants performed it, and there was an optional break between them.

The change signal task consisted of 6 blocks of 107 trials each. (The trial count was selected for consistency with the Brown and Braver (2005) experiment.) After each block, subjects were allowed to take a brief break. A diagram of the possible sequences of events during a trial and their probabilities is shown in Figure 1. At the start of each trial, a cue was presented in one of two colors, which was associated with either a high error condition or a low error condition. The significance of the colors in the task was not explained to the participants, but they were made aware that there would be two colors, just as in Brown and Braver (2005).

After 1 second, the cue was replaced with an arrow in the same color that pointed right or left. The participant was instructed to respond to this “go signal” with the appropriate arrow on the keyboard. On 1/3 of the trials, a larger arrow pointing in the opposite direction appeared after a brief delay (the “change signal delay,” or CSD). The participant was instructed that, in these circumstances, they should inhibit their initial response and instead respond to the larger arrow. The larger arrow, or “change signal,” always pointed in the opposite direction as the go signal.

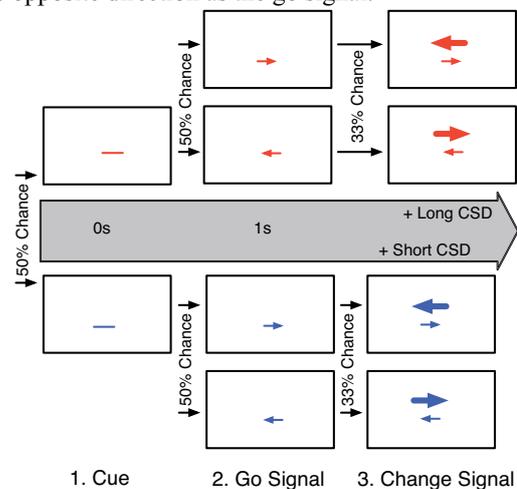


Figure 1: State diagram of a change signal trial. Error condition and arrow direction appear with equal probability. Change signals only appear on 33% of the trials.

The change signal delay was dynamic, and varied according to the error rate condition. At the start of the task, the CSDs for both conditions were set to 250ms. For the high error rate condition, a correct response increased the CSD by 50ms, while an incorrect response decreased it by 50ms. The low error rate CSD behaved similarly, except that it only increased by 2ms when a correct response was made. These manipulations replicate Brown and Braver (2005), and were designed to produce different error rates in

the two conditions. For both conditions, the CSD was constrained between 20 and 800ms. Each trial allowed responses up to a full second after the go signal was displayed. If no response was provided within that time, the trial was recorded as a non-response and another trial was automatically initiated.

Halfway through the trials for the change signal task, the mapping between color cue and error condition was reversed, although the CSDs for each condition were not reset. After the participant completed the experiment, he/she was asked whether they noticed the role of the colors in the experiment. In doing so, the experimenter was testing whether the participant acquired any explicit knowledge. We considered it adequate evidence if the respondent indicated that one color was faster or more difficult than the other.

The 2AFC task was identical to the change signal task in every respect, except that no change signals were presented.

Results

Generally speaking, the results from our study were consistent with the Brown and Braver (2005) study. The aggregate data for the change signal task shows the expected slowing in reaction time as the experiment progresses. Conversely, the 2AFC task shows slightly improved reaction times over the duration of the experiment when reaction time is regressed against trial index ($b = -.032$, $R^2 = .00040$, $F(1,20342) = 82.49$, $p < .001$; see Figure 2). This result argues against time-on-task based declines in cognitive performance as the source of slowing in the change signal task, and supports the hypothesis that participants were strategically “hedging” their response times. Furthermore, an ANOVA with factors of block and error likelihood confirms that the response times between the two error conditions are significantly different, ($F(5,17552) = 62.48$, $p < .001$), as found in our previous research (Moore et al., 2010). Overall, participants made errors on 34% of the trials in the high error rate condition, and on 5% of the trials in the low error rate condition.

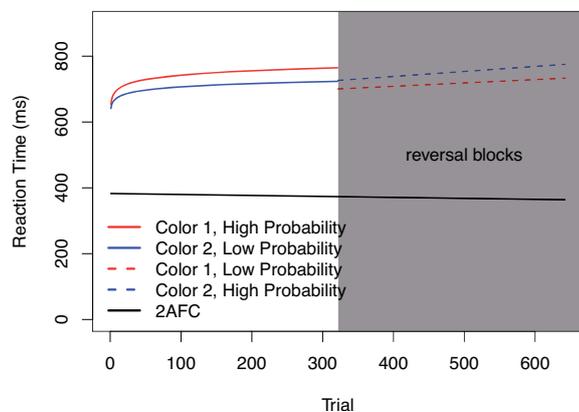


Figure 2: Regression lines for aggregate participant data, showing the strategic hedging over time, as well as the significant disparity across the two conditions.

A more revealing perspective on these results can be observed in Figure 3, which breaks apart trials where a change signal was presented (change condition) versus trials where no change signal was presented (go condition). The experimental condition permutations then become:

1. go-low: go signal only (no change signal) presented in the low error condition color,
2. go-high: go signal only (no change signal) presented in the high error condition color,
3. change-low: change signal present in the low error condition
4. change-high: change signal present in the high error condition.

The figure also includes the relative reactions times in the 2AFC task. All reaction times are measured from the onset of the go signal, and results are aggregated across blocks.

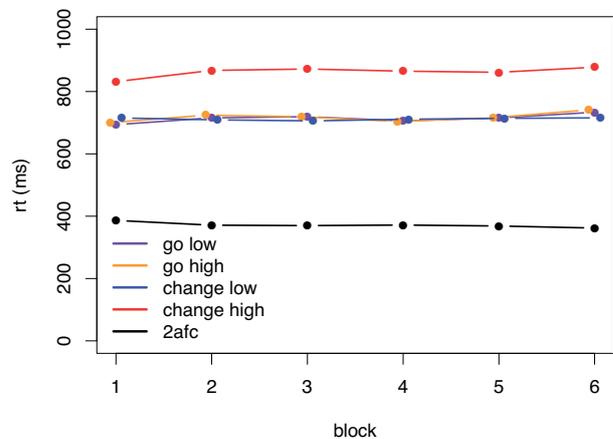


Figure 3: Mean reaction times aggregated by block for correct responses in each of the four change signal task conditions as well as the 2AFC task. Reaction times are measured from the onset of the go signal.

Notice that the 2AFC response times are substantially faster than the response times in the change signal task. As mentioned previously, our theory proposes that this is strategic; participants are hedging their responses to go signals in order to allow for the possibility of a change signal being presented. Under this account, the go conditions represent the situation where individuals exhaust their hedge time and produce a response. Therefore, the difference between the 2AFC and go condition reaction times would represent the mean participant hedge time, which is about 300ms.

Also notice the disparity between the change-high and change-low conditions in Figure 3. In Moore and Gunzelmann (2010) we suggested that implicit association of stimulus color might explain the disparity. This was also supported by the original Brown and Braver (2005) work, which focused on learned responses to error conditions in the anterior cingulate cortex (ACC). However, if there is learning in the ACC, it is not reflected in the empirical data because the stimulus color only impacts response time on trials where a change signal is presented. There is no

difference in RT between the two “go” conditions ($F(1,12607)=.12, p=.73$). If participants were learning the association between error-likelihood and stimulus color, we would expect to see an identical disparity between the go-low and go-high conditions.

Rather than implicit learning, our finding suggests that the emergent difference in reaction times between the change-high and change-low conditions can be explained as an artifact of the task itself. Recall that correct responses to change signals increase the change signal delay by 2ms in the low error condition and 10ms in the high error condition. The different step functions result in change signal delays that tend to be shorter in the low error condition than in the high error condition. Regardless of the error condition, however, participants respond immediately to change signals that appear within the hedge period. Thus, reaction times for the low error condition are faster than the high error condition because of the shorter change signal delay.

To demonstrate that change signal delays are driving the difference in reaction times across the two change conditions, Figure 4 removes the change signal delay from change condition reaction times. (i.e. Reaction times are now measured from the onset of the change signal for the trials with a change signal). The disparity between the high and low change conditions in Figure 3 is greatly reduced, which reinforces the position that it is an artifact of the task itself. In fact, in this analysis response times are slightly *faster* in the high error likelihood condition.

A descriptive analysis of the response distributions, as well as quantitative evidence of a disparity in lapses across the two conditions (14% for the change-high condition versus 1% for the change-low condition), both suggest that the remaining discrepancy in reaction times may be accounted for by a truncation of the response distribution in the change-high condition. This truncation occurs because some of the response times are very close to the 1-second trial time limit.

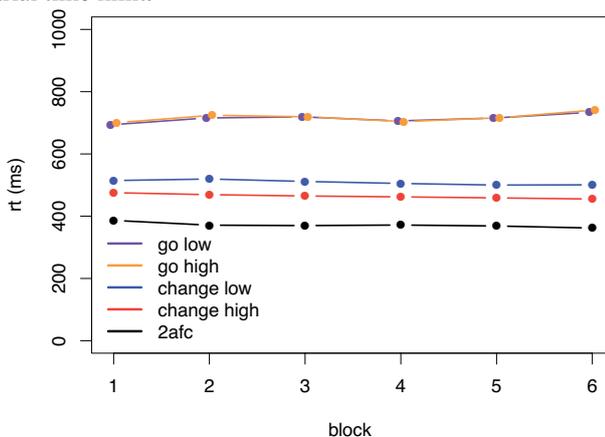


Figure 4: Mean reaction times aggregated by block for correct responses in each of the four change signal task conditions as well as the 2AFC task. Reaction times are measured from the onset of the go signal for go conditions, and the onset of the change signal for change conditions.

Another item in Figure 4 that warrants attention is the fact that both change condition reaction times are higher than the 2AFC reaction times. In all three of these conditions participants can respond immediately to the stimulus, which suggests that the change conditions impose an extra cognitive penalty. This is an important consideration for the model, which will be discussed in the following section.

The Change Signal Model

The change signal model was developed within the Adaptive Control and Thought – Rational (ACT-R) cognitive architecture (Anderson, 2007). ACT-R is a symbolic production system coupled with mathematically grounded mechanisms that reflect sub-symbolic influences. The change signal model is instantiated within the architecture by supplying knowledge in the form of production rules and declarative chunks. Our model is relatively simple, consisting of 14 productions and no initial declarative knowledge. We characterize it as a “procedural” model because it does not rely upon declarative retrievals to function.

At a high level, the critical feature of the model is a strategic delay of its response to the stimulus to accommodate the possibility of a change signal. As described in the task section, we refer to this as the hedge time. If a change signal does occur during the hedge time, the model generates a response to the larger arrow as soon as it appears (change signal). If no change signal occurs during the hedge time, the model responds to the original arrow (go signal). Time estimates are noisy, and are derived using a mechanism proposed by Taatgen, van Rijn, and Anderson (2007).

The model will also adjust its hedge time (which is maintained as a slot in the goal chunk) dynamically based on responses to change signals. When a change signal is detected after it has already responded to a go signal, the hedge time is increased in hopes that it will correctly catch the change signal in the future. When the model does correctly respond to a change signal, or when the model sees an unexpected cue because it failed to respond before the trial expired, the hedge time is decreased.

The 2AFC task is identical to the change signal task except no change signals are ever presented. As a result the model can perform the 2AFC task unaltered using a subset of the full procedural knowledge; those productions involved with responding to change signals never fire.

As discussed in the previous section, Figure 4 shows that change signal responses incur an extra penalty in response time. There are several plausible theories to account for the increased response time when a change signal is encountered. In our model, the delay is attributed to motor control. The model prepares its response to the “go” signal when it is presented. Thus, when a change signal is observed, the motor system is reset, which negates the preparation that was done. To respond, the model must first

plan the motor movements and then execute them. The extra planning adds time to the response process.

Model Fitting

The initial hedge time, hedge increase, and hedge decrease are tunable parameters in the model. We also chose to adjust the ACT-R default action time, which describes the speed of the average production cycle.

Because the model shares knowledge between the 2AFC and change signal tasks, it was fit in two stages. The 2AFC task was fit first because it requires fewer degrees of freedom. The empirical data supports the conclusion that participants did not engage in strategic hedging while performing the 2AFC, so initial hedge time, hedge increase, and hedge decrease parameters are all set to 0.

The only remaining parameter to fit in the 2AFC task is default action time (DAT). It defaults to 50ms, but Stewart, Choo and Eliasmith (2010) have shown that tasks composed of simple productions (such as those implemented to perform the 2AFC and change signal tasks) will have shorter cycle times. This is also consistent with our work with the psychomotor vigilance task (PVT), where we typically find default action time values of approximately 40ms (Gunzelmann, Moore, Gluck, Van Dongen & Dinges, 2010). Rather than refitting the 2AFC task independently, we chose to constrain DAT to the value obtained from the PVT, resulting in the dashed black line in Figure 5. This 40ms value was held constant while fitting the model with the change signal task, as well.

The remaining parameters include the initial hedge time, the increase in hedge time when the model detects an error, and the decrease in hedge time when the model correctly responds to a change signal. All three parameters are specific to the hedging strategy, and are not general parameters of cognition. (We also enabled a mechanism to provide some stochasticity to production cycle times, but left that parameter at its default value.) To resolve the three dimensional parameter space, we used large scale computational resources running our in-house search software (Moore, 2011). The model was then rerun using the predicted optimal values to produce the change signal model results in Figure 5.

Results

The overall RMSD for the block-aggregated data across all 5 conditions in Figure 5 was 33.4ms. The simpler 2AFC task fit the best at 15.9ms RMSD, while the go-high condition fit the worst, at 50.1ms.

The model was not very sensitive to the initial hedge time parameter. Its behavior was primarily driven by the hedge-up and hedge-down values, as they are critical to establishing and maintaining the equilibrium between the model and task. The optimal values (27% increase when hedging up, and 6.5% decrease when hedging down) suggest that participants were more liberal with hedging up (waiting longer when an error is detected) than they were hedging down (responding sooner when a change signal is

correctly detected). Furthermore, there was clearly a relationship between the two variables: larger upward hedges could be compensated by larger downward hedges to maintain a reasonable fit. A degree of freedom could potentially be reduced if one of the two parameters could be experimentally isolated.

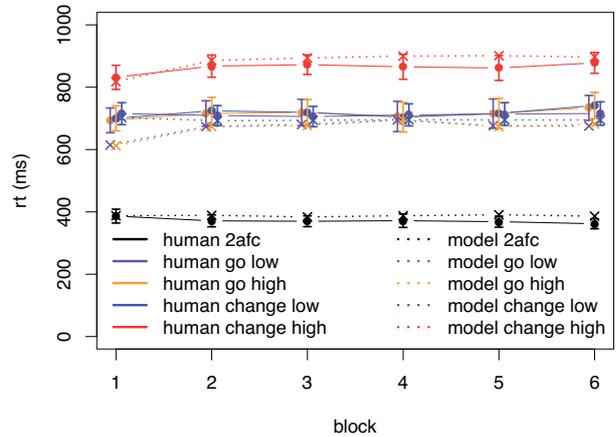


Figure 5: Model fits to empirical data for correct responses.

In addition to the reaction time across blocks, there are several other statistics that can be examined to evaluate the model's performance relative to human participants. The percent of correct responses, the standard deviation in reaction time, and the number of non-responses are three measures shown in Figure 6. The model's performance was within the inter-quartile range on all three measures, and it performed particularly well with percent correct and proportion of non-responses.

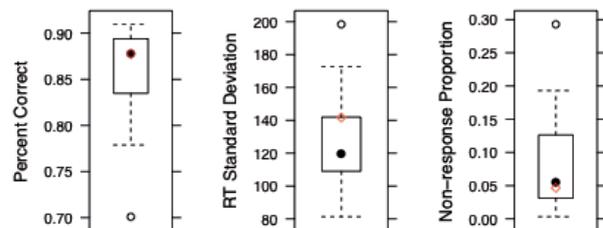


Figure 6: Secondary measures of fitness in the change signal task. The box and whiskers demonstrate the variation across subjects, while the red diamond indicates the mean for the model.

Conclusion

In Moore et al. (2010) a cognitive model of the Change Signal task raised some important questions that could not be fully addressed with the available data. The follow-up study reported in this paper provided an opportunity to further solidify our understanding of the cognitive mechanisms associated with the task.

A primary benefit was having performance data from participants for both the 2AFC and change signal tasks, for

several reasons. First, participants performed the 2AFC consistently and much more quickly than they did in the change signal task. This supports our theory of a hedging strategy to manage change signal responses. The lack of performance degradation in the 2AFC task shows that within task fatigue plays a very small role, if any. Furthermore, participant hedging appears to be dynamic, as is evidenced by the slowing reaction time throughout the session. This informs the model, and justifies the model's hedging parameters.

The dual task / repeated measures design of the experiment also allowed us to isolate one of the parameters in the change signal model (default action time) and fit it independently within the context of the 2AFC. It was an unexpected additional benefit that we were able to use a value for that parameter derived from previous research using a different task. The remaining parameters, which were all related to hedging, could then be fit within the context of the change signal task.

The change signal model has demonstrated that it can perform two tasks with overlapping knowledge using a single set of model parameters. In doing so, it inspires confidence in the theory that the model represents. We believe this is well within the spirit of looking beyond a simple model fit for validation (Salvucci, 2010).

Acknowledgments

The views expressed in this paper are those of the authors and do not reflect the official policy or position of the Department of Defense, the U.S. Government, or L3 Communications. This research was sponsored by grants 07HE01COR and 10RH04COR from the Air Force Office of Scientific Research. It was also supported through the Repperger Internship Program sponsored by the 711 HPW/RH Chief Scientist's Office.

References

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford University Press.

Brown, J. W., & Braver, T. S. (2005). Learned predictions of error likelihood in the anterior cingulate cortex. *Science*, *307*(5712), 1118. doi:10.1126/science.1106642

Gobet, F., & Ritter, F. E. (2000). Individual data analysis and Unified Theories of Cognition: A methodological proposal. In *Proceedings of the 3rd International Conference on Cognitive Modeling* (pp. 150-157). Veenendall, The Netherlands: Universal Press.

Gunzelmann, G., Byrne, M. D., Gluck, K. A., & Moore, L. R., Jr. (2009). Using computational cognitive modeling to predict dual-task performance with sleep deprivation. *Human Factors*, *51*(2), 251-260. From <http://handle.dtic.mil/100.2/ADA504480>.

Gunzelmann, G., Moore, L. R. Jr., Salvucci, D. D., & Gluck, K. A. (2011). Sleep loss and driver performance: Quantitative predictions with zero free parameters. *Cognitive Systems Research*, *12*, 154-163.

Gunzelmann, G., Moore, L. R., Jr., Gluck, K. A., Van Dongen, H. P. A., & Dinges, D. F. (2010). Fatigue in sustained attention: Generalizing mechanisms for time awake to time on task. In P. L. Ackerman (Ed.), *Cognitive fatigue: Multidisciplinary perspectives on current research and future application* (pp. 83-96). Washington, DC: American Psychological Association.

Halverson, T., & Gunzelmann, G. (2011). Visual search versus memory in a paired associate task. In *Proceedings of the Human Factors and Ergonomics Society 55th Annual Meeting* (pp. 875-879). Thousand Oaks, CA: SAGE. (DOI: 10.1177/1071181311551182)

Logan, G. D. & Cowan, W. B. (1984). On the ability to inhibit thought and action: A theory of an act of control. *Psychological Review*, *91*, 295-327.

Meyer, D. E., & Kieras, D. E. (1997). A computational theory of executive cognitive processes and multiple-task performance. Part 1: Basic mechanisms. *Psychological Review*, *104*(1), 3-65.

Myers, C. W. (2009). An account of model inspiration, integration, and sub-task validation. In *Proceedings of the 9th international conference on cognitive modeling* (pp. 333-335). Manchester, United Kingdom.

Moore, L. R., Jr. (2011). Cognitive model exploration and optimization: A new challenge for computational science. *Computational. Computational & Mathematical Organization Theory*, *17*, 296-313.

Moore, L. R., Jr., Gunzelmann, G., & Brown, J. W. (2010). Modeling statistical learning and response inhibition with the change signal task. In D. D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the 10th international conference on cognitive modeling* (pp. 169-174). doi:10.1126/science.1105783

Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review*, *107*, 358-367.

Rosenbloom, P. S., Laird, J. E., and Newell, A. (1993). *The SOAR Papers*. Cambridge, Massachusetts: MIT Press.

Salvucci, D. D. (2010, April). *Cognitive supermodels*. Invited talk, European ACT-R Workshop, Groningen, The Netherlands.

Stewart, T. C., Choo, X., & Eliasmith, C. (2010). Dynamic behavior of a spiking model of action selection in the basal ganglia. In D. D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the 10th international conference on cognitive modeling* (pp. 235-240). Philadelphia, PA: Drexel University.

Taatgen, N. A., Rijn, H. v., Anderson, J. R. (2007). An integrated theory of prospective time interval estimation: The role of cognition, attention, and learning. *Psychological Review*, *114*(3), 577-598.

Verbalization of problem solving processes in unaided object assembly

Linn Gralla (gralla@uni-bremen.de)

Department of Linguistics and Literary Sciences, Universität Bremen, Germany

Thora Tenbrink (tenbrink@uni-bremen.de)

Department of Linguistics and Literary Sciences, Universität Bremen, Germany

Keywords: cognitive discourse analysis, problem solving, cognitive processes, linguistic representation

Introduction

Imagine buying a dollhouse for your niece at a garage sale, and what you get is a set of wooden pieces and a picture of the house, but no instruction manual. How do you solve the problem of assembling the pieces to build the house? What you are facing is a well-defined problem, since you know the goal state and all objects needed to reach the desired state; only the correct sequence of actions is missing.

Newell and Simon (1972) postulate the following general structure when a problem is encountered: *orientation phase*, *construction of the problem space*, and *exploration of the problem space by selecting and applying operators*. During the orientation phase the problem is recognized and the situation is analyzed. This initial analysis is extended by the construction of a detailed representation of the problem (*problem space*) that includes information on the initial state, the operators that can be applied to change this state, and information on how the goal state is defined. The construction may be based on the analysis of the task environment or retrieved from long-term memory. The problem solving process itself is defined as a *search process* through the problem space. In the search process the problem solver applies different methods to create new states and checks repeatedly if those qualify to be the goal state. Palmer (1977:466) considers the following processes crucial to organizing problem parts: *exploratory hypotheses*, *false leads*, *dead ends*, *backtracking*, and *fresh starts*.

Unaided object assembly: an explorative study

In our study we aim to identify the cognitive processes involved in unaided object assembly by examining think-aloud protocols, along with a better understanding of how these processes are expressed in language. Think-aloud protocols are traditionally used to gain insights on cognitive processes involved in problem solving (Ericsson & Simon, 1993), typically focusing on content, i.e., *what* is verbalized. Further insights can be gained from analysis of the language used, i.e. *how* thoughts are verbalized. Roth (1985), for example, showed that unsuccessful problem solvers used more negations, adversative conjunctions, and modals than successful problem solvers. Caron-Pargue and Caron (1991) illustrate how linguistic markers (e.g. lexical choice, connectives) give insight on the problem solver's representation with regard to organization, function, and change.

Design

56 university students (26 male, 30 female; aged 19-42 years, mean age 24 years) participated in this study for course credit or monetary compensation. They were told that they would be given object parts that need to be assembled without a manual. Knowledge of the goal state varied between mention of "a dollhouse", being shown a picture of the assembled dollhouse, and no such information. Here we focus on phenomena common to all three conditions. After the instruction was given, participants entered a room and saw a cardboard box and a triangular piece of wood on a table. The box contained 13 wooden parts. Participants were instructed and reminded to think aloud while solving the task.

Analysis

30 think-aloud protocols were analyzed for current purposes, namely the identification of general problem solving processes and their expression in language.

First, the general structure of the problem solving process was identified by a detailed content analysis of 11 protocols. With regard to the *search process*, the process categories as proposed by Palmer (1977) were identified and extended. These categories were linguistically analyzed in 18 protocols with regard to verb classes (cf. Halliday & Matthiessen, 1999), conjunctions, negations and affirmations, and the discourse markers *so* and *okay*. Next, all 30 protocols were annotated according to these categories in order to describe their distribution in more detail, and to identify recurrent sequences of processes.

Results

All protocols showed the general structure of an introductory sequence in which parts of the instruction were repeated or object parts were recognized, and first associations were verbalized. The main body of the protocols consisted of the actual problem solving process. In most protocols the task was concluded by a brief evaluation of the assembled object, or the personal skills in solving the task. Inspired by Palmer's (1977) approach, *hypotheses*, *false leads*, *dead ends*, and *fresh starts* were identified. The content analysis revealed four additional categories, namely *description of mental state*, *perception of object features*, *action* (including plans for action), and *positive evaluation*. Since *false leads* can also be understood as evaluations, this category was renamed *negative evaluation*.

Altogether, 1,405 processes were identified and annotated in the 30 protocols. Of these, *hypotheses* were most frequent (42.5%), and 20.5% were instances of *action*. *Evaluations* were *positive* (9.7%) or *negative* (7.8%); these will be combined in the following.

The chain *hypothesis–evaluation* was found in 42.5% of all possible process chains starting with *hypothesis*, and *hypothesis–action* in 41.5%. For the category *action*, the chain *action–hypothesis* was most frequent (56.1%). Those three sequences occurred in 26 out of 30 protocols. The sequence *action–evaluation* accounted for 35.0% of all possible chains starting with *action*. *Positive evaluation–hypothesis* occurred in 46.3% of all chains starting with a positive evaluation. These two chains were identified in 23 and 24 protocols, respectively. *Negative evaluation* led to a *hypothesis* in 61.3% of cases; however, this chain occurred in only 11 protocols. The combination of these sequences in the four-process-cycle *hypothesis–action–evaluation–hypothesis* was identified in nine protocols.

Based on a detailed analysis of 18 protocols the following linguistic markers were identified. The category *hypothesis* was characterized by frequent occurrences of verbs of ‘being and having’ (62.2% of all such verbs belonged to the category *hypothesis*), as well as verbs denoting mental processes (44.1%), e.g. *think* or *believe*. Almost half (47.1%) of the utterances in this category were connected by conjunctions; mainly introducing reasons using *because* (41.1%). Further re-occurring markers of *hypotheses* were short phrases expressing the mental state (*I think*) or mental activities (*I’m asking myself*) of the problem solver. The category *action* was characterized by verbs of ‘doing and happening’, such as *put* or *assemble* (66.1%). Here, the connectives *and* and *because* occurred in 25.7% of cases. The discourse markers *so* and *okay* were identified in 52.0% of all utterances classified as *positive evaluation*, with *so* (79.6%) more frequent than *okay*. Furthermore, this category contained 75.0% of all affirmative words, such as *right* or *super*. Most expressions classified as affirmation or negations were found in the category *negative evaluation*, with 72.3% of all utterances containing such an expression. Almost all of those occurrences were negations, such as *nee* (98.5%). On the other hand, 31.9% of all negations occurred in the category *hypothesis*.

Discussion

The following picture emerges when comparing the problem solving processes identified in our protocols to those proposed in the literature. Content analysis of the introductory sequence of the protocols revealed that it contains the *orientation phase* and the *construction of the problem space* as described by Newell and Simon (1972) because participants were found to recall instruction details and start exploring the task environment. Schoenfeld (1985) also identified *read* and *explore* as the episodes in which a problem solver engages first. The main body of the protocols included the categories *hypothesize* and *action* that correspond to the processes of selection and application of

operators respectively. *Hypotheses* represent verbalizations of possible states, such as concepts and object configurations. The reasoning process about these possible states is illustrated by the frequent occurrence of *because* in this category. The continuous evaluation of newly derived states is expressed in the categories *positive* and *negative evaluation*. As *so* conveys a meaning of result (Schiffrin, 1988) *positive evaluations* can be interpreted as signals for reaching sub-goals. This stands in contrast to negations, so-called markers of denial, which signal the rejection of an idea and may result in a complete reorganization of the representation (Caron & Caron-Pargue, 1991:32). The finding that *evaluations* are frequently followed by a new *hypothesis* supports these interpretations. Both findings illustrate the importance of the control process.

Empirical research revealed the difficulty of identifying longer process sequences since, unlike in theoretical models, the processes tend to occur in various sequential orders (e.g. Wedman et al., 1996). In our study, we found that *actions* frequently occurred with *hypotheses* and *evaluations*. A combination of these processes, namely *hypothesis–action–evaluation–hypothesis*, was identified in one-third of all protocols. This sequence represents the theoretically assumed progression of problem solving processes that is repeated until the goal state is reached.

Acknowledgments

Funding by the Zentrale Forschungsförderung Universität Bremen is gratefully acknowledged.

References

- Caron-Pargue, J. & Caron, J. (1991). Psychopragmatics vs. sociopragmatics: the function of pragmatic markers in think aloud protocols. In Verschueren, J. (ed.). *Pragmatics at issue – Selected papers of the International Pragmatics Conference*. Amsterdam: Benjamins, 29-36.
- Ericsson, K.A. & Simon, H.A. (1993). *Protocol Analysis – Verbal reports as data*. Massachusetts (USA): MIT Press.
- Halliday, M. A. K & Matthiessen, C. M. (1999). *Construing Experience Through Meaning – A Language Based Approach to Cognition*. London: Continuum.
- Newell, A. & Simon, H.A. (1972). *Human Problem Solving*. N.J.: Prentice-Hall.
- Palmer, S.E. (1977). Hierarchical Structure in Perceptual Representation. *Cognitive Psychology* 9, 441-474.
- Roth, T. (1985). Sprachstatistisch objektivierbare Denkstilunterschiede zwischen ‚guten‘ and ‚schlechten‘ Bearbeitern komplexer Probleme. *Sprache & Kognition* 4, 178-191.
- Schiffrin, D. (1988). *Discourse markers*. Cambridge: Cambridge University Press.
- Schoenfeld, A. H. (1985). *Mathematical Problem Solving*. Orlando: Academic Press.
- Wedman, J., Wedman, J., & Folger, T. (1996). Analysis of analogical problem-solving processes via think aloud protocols. *Journal of Research and Development in Education*, 30 (1), 51-62.

Modeling Risky Decision Making by Cellular Automata

Kenryo Indo (Kindo@Kanto-Gakuen.Ac.Jp)

Department of Business Administration, 200 Fujiyagu

Ota, Gunma 3738515 Japan

Keywords: risky decision making, violation of stochastic dominance, ambiguity aversion, cellular automata.

Introduction

This paper proposes a new approach to the problem of decision making under risk (and partly under uncertainty) by a simple cellular automaton which can simulate well-known anomalies of risky choice. This field has been studied mainly by psychologists, economists, management scientists, and more recently neuroscientists so far. Various types of anomalous choice patterns which violate the prediction according to the probability theory and the expected utility theory are called paradoxes, or anomalies. Specifically, descriptive theory of risky choice should model the effects of event-splitting (and event-merging) to violate the first-order stochastic dominance, as well as the ambiguity aversion and the common consequence effect.

By editing branches and matching of rows in two gambles, one could invite anomalous choice which reverses mathematically the same choice, therefore decision maker's risk cognition permit a manipulation by the problem representation. Figure 1 and Figure 2 show examples (They are the choice problems 2 and 3.2 in Birnbaum (2008)).

<p>Gamble A</p> <p>90% win \$96</p> <p>5% win \$14</p> <p>5% win \$12</p>	<p>Gamble B</p> <p>85% win \$96</p> <p>5% win \$90</p> <p>10% win \$12</p>
---	--

Figure 1: Choice problem between A and B.

<p>Gamble C</p> <p>85% win \$96</p> <p>5% win \$96</p> <p>5% win \$14</p> <p>5% win \$12</p>	<p>Gamble D</p> <p>85% win \$96</p> <p>5% win \$90</p> <p>5% win \$12</p> <p>5% win \$12</p>
--	--

Figure 2: Choice problem between C and D.

Note that Gamble A is the same as Gamble C if two of the best branches (85% and 5% of winning \$96) are combined. Conversely, an event-splitting for the first branch of Gamble C generates 4-branch Gamble A. Similarly, Gamble B is made by an event-splitting for the last branch (the worst) of Gamble D. Gamble D is first-order stochastically dominated by Gamble C (i.e., the graph of C's cumulative probability is always not above than of D and strictly below at some state). However, many of human subjects choose A as reported in the experimentation study (The choice ratio are 73% and 6% for B and D).

Violation of stochastic dominance is a type of anomaly which cannot be explained by either Expected Utility Theory (EUT) or Cumulative Prospect Theory (CPT). Recently, Birnbaum's Transfer of Attention Exchange (TAX) theory is rediscovered to explain the event-splitting effect as well as other new paradoxes.

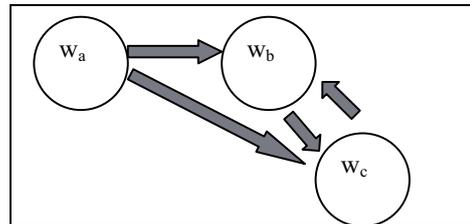


Figure 3: A three-branch TAX theory (Birnbaum (2008), p.471). Assume a DM who prefers 'a' to 'b' to 'c' and the probability weight $W(p) = p^{0.7}$ with 25% transfer along each arrow. Then, this model can predict a reversal in comparing between Gamble E = (\$100, 80%; \$50, 15%) and Gamble F = (\$100, 95%; \$7, 5%) by a simultaneous split of the probabilities of \$50 in E and of \$100 in F into 10% and 5%.

Two questions could have been pointed out. First, the TAX model proposed in order to predict event-splitting effects has the maximal reversal effect at simultaneous bisections for the best branch of originally inferior gamble and the worst branch of superior one. This may generate also a counterintuitive example (Namely, Gamble E' = (\$100, 80%; \$50, 7.5%; \$50, 7.5%) and Gamble F' = (\$100, 47.5%; \$100, 47.5%; \$7, 5%). Second, the theory lacks a consistent explanation of modularity-switching for any probability weight function. In order to model ambiguity aversion should be super-modular, while the function of event-splitting effect should be sub-modular.

In this paper, an alternative approach to modeling cognitive dynamics of risky choice by using cellular automata is proposed, and computational experiments which generate the anomalies of risky choices are demonstrated.

Cellular Automaton Model

Cellular Automaton

A cellular automaton is a collection of cells live in a grid (or torus) normally a Cartesian of some number of lines (or loops) each of which consists of a finite number of cells. Each cell has its own permissible state from a given set of numbers (or colors). Transition of states for each cell is governed by a given set of rules which depend on the state profiles of the neighbors.

The state of a cell changes over time according to the rule which may depend on the current state, and possibly on the previous states, of neighboring cells. The computer simulation of cellular automata mimics such processes iteratively. The shape as a whole is often complex even if the governing rules are simple, however, ordered patterns may emerge unpredictably. It is known that cellular automata have a capability of producing the universal Turing machine. (For more detail see Wolfram (2002).)

Stochastic Local q-Majority Vote

In order to model cognitive dynamics of risky choice and to simulate anomalies mentioned in the preceding section, we will devise the following two-dimensional cellular automaton which represents working memory of the decision maker.

- The universe U is a two-dimensional torus with the index modulo 10, consisting of cells $c(X, Y)$, where $X, Y = 1, 2, \dots, K$. We assume $K = 10$ for the sake of practical computation.
- The neighborhood over which cells affect one another is a square which consists of nine cells including the center cell (i.e., the Moore neighborhood).
- A possible state of each cell at each step is assumed to be a triple of bits, i.e., eight is the number of permissible states for each cell at any time. Note that the three-bit pattern can represent any (in)transitive ordering of three objects, for example, a 110 stands for $x > y > z$. The first bit is a value 1 if x is preferred to y by the DM, else 0. The second and the third bits index the paired comparisons of y to z and of z to x respectively.
- A stochastic quotient Local Majority Vote is assumed to be *the rule of influence* which governs behavior of each cell affected from its neighborhood as the set of voters of the local polls for each step. The number q is assumed to be $2/9$ (or $3/9$) in the following computer simulation experiments.

The intuitive interpretation of cells are linearly matched two gambles each of which is decomposed into a hundred of “one-percent” branches. The first assumption reflects the key factor in this model is the sequences and spatial configuration of the matching two gambles --- a metaphor of battle on sphere which to decide whether gambles wins.

Local Majority Vote was first argued by Peleg (2000) in the context of algorithm in distributed computer network. It also seen as a version of the ACM (Axelrod, 1997) modified the neighbor similarity criteria by q -local majority vote. The stochastic q -version of LVM using the above assumptions produces various tiled patterns if $q = 3/9$ (Indo, 2012).

In each step this procedure randomly selects a neighbor (the focal member) of each cell (the center of neighborhood) and set polls for each bit by a q -majority vote in the neighborhood. That is, for each bit the center cell should adopt the focal member’s evaluation if the counted number of neighbors who have the same bit value as the focal member exceeds $9q$.

Simulation Results

This section summarizes computer experimentations of the cellular automata described in the previous section which can simulate the anomalies of risky choice mentioned in the first section by using *artisoc* a toolkit of agent-based modeling.

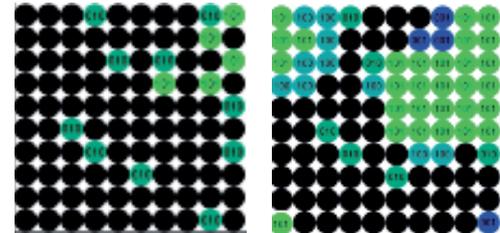


Figure 4: A cellular automaton of gamble comparison.

Figure 4 shows a cellular automaton which evaluates the two-branch gamble comparison described in Figure 3. In the left side of Figure 4 is the initial state of torus. A small colony of five Cell 101s represents five times of a match of two unit branches (\$50, 1%) from Gamble E and (\$7, 1%) from Gamble F. In the right side of Figure 4 depicts a tentative state where the Cell 101 colony is evolving to replace “black” circles each of which is Cell 000 a tie of (\$100, 1%). The count of first bit indicates that E wins as a whole, and the remaining two bits record difference. Note that the ten Cell 010s, each of which stands for a match (\$50, 1%) from E v.s. (\$100, 1%) from F, are dispersed in the ground of eighty five Cell 000s. For many event-splitting that make unmatched event branches such like as E’-F’ have similar configuration. However, a crowd rearrange of the Cell 010s could make F a final choice. For more detail see the poster and handout.

Acknowledgments

The author acknowledges support from Kanto Gakuen University (grant Gakujutu Kenkyu Hi in 2011). And the author gratefully acknowledges use of *artisoc* from the academic program by Kozo Keikaku Engineering Inc.

References

- Axelrod, R. (1997): The Dissemination of Culture: A Model with Local Convergence and Global Polarization. *Journal of Conflict Resolution*, 41 (2), 203-226.
- Birnbaum, M. H. (2008). New paradoxes of risky decision making. *Psychological Review*, 115(2), 463-501.
- Peleg, D. (1996). Local Majority Voting, Small Coalitions and Controlling Monopolies in Graphs: A Review, In N. Santoro & P. Spirakis (Eds.), *Structure, Information and Communication Complexity: 3rd Colloquium, SIROCCO '96: Proceedings* (pp.152-189). Carleton University Press.
- Indo, K. (2012). A Multi-Agent Simulation of Preference Formation by Local Majority Vote, mimeo.
- Wolfram, S. (2002). *A New Kind of Science*. Champaign, IL: Wolfram Media.

Conditioning for Least Action

Eduardo Alonso (E.Alonso@city.ac.uk)

Department of Computing, Northampton Square
London, EC1V 0HB United Kingdom

Michael Fairbank (abdy934@soi.city.ac.uk)

Department of Computing, Northampton Square
London, EC1V 0HB United Kingdom

Esther Mondragón (e.mondragon@cal-r.org)

Centre for Computational and Animal Learning Research
St Albans AL1 1RQ, United Kingdom

Abstract

It is well known that, in one form or another, the variational Principle of Least Action (PLA) governs Nature. Although traditionally referred to explain physical phenomena, PLA has also been used to account for biological phenomena and even natural selection. However, its value in studying psychological processes has not been fully explored. In this paper we present a computational model, value-gradient learning, based on Pontryagin's Minimum Principle (a version of PLA used in optimal theory), that applies to both classical and operant conditioning.

Keywords: Value-gradient learning; conditioning; behavior systems; bliss point; optimality; principle of least action.

The Principle of Least Action

Of all the possible trajectories a ball thrown into the air can follow why does it follow one in particular, a parabola? Why doesn't it go up, stay a while at its highest point and then fall down? On the one hand, the ball wants to spend a lot much time near the top of its trajectory since this is where the kinetic energy is least and the potential energy is greater. On the other hand, if it spends too much time near the top, it will really need to rush to get up there and get back down and this will take a lot of action. The perfect compromise is a parabolic path. In physical parlance, the true dynamical trajectory of the ball is the one that makes the action "least" (actually stationary).

Formally, the action to be minimized is the integral of a function, the Lagrangian, over time. The Lagrangian itself describes completely the dynamics of the system under consideration as the difference between its kinetic energy (the energy due to the motion, how much is "happening") and its potential energy (the energy due to its position or configuration, how much "could happen"). In short, Nature is as lazy as possible: the ball follows a particular trajectory not because of the effect of gravitation *per se*, but because it "minimizes" action. In fact, this condition is equivalent to the Euler-Lagrange equation of motion that encapsulates the Principle of Least Action and that, when transformed into its Hamiltonian form, reflects the symmetries of Nature. These are fundamental concepts upon which modern Physics is based.

The question is, can we export this variational analysis to the study of learning and behaviour?

Optimization in Classical Conditioning

Let's consider acquisition of an eye-blink conditioned response when a light is paired with a mild shock: at first the likelihood of a response to the light is low because of the absence of prior light-shock pairings. There is then a rapid increase in magnitude of the response, which diminishes gradually as training progresses until there are no further increases in the measure of the conditioned response. The shape of the learning curve is typical of that found in many studies of conditioning. How is this pattern of behaviour explained? Why don't animals learn "all" in a single trial? Or learn rapidly at the beginning, then stop and then learn again? In a way, we are facing the same questions as we did when considering ball trajectories. And it is paramount that we answer them since *conditioning is at the basis of most learning phenomena and thus of animal cognition.*

More generally, classical conditioning refers to the type of learning that occurs when pairing two stimuli, typically an originally neutral stimulus (say a tone or a light) and an unconditioned stimulus (US), that is, a stimulus that is biologically relevant to the animal (for instance, food) that elicits an automatic or unconditioned response (UR, for example, salivation). If this pairing is repeated over time, the animal will learn to anticipate the US and start responding to the signal, the neutral stimulus. The neutral stimulus will become a conditioned stimulus (CS) and trigger a response (CR, typically the UR itself).

In order to explain this type of phenomena, Rescorla and Wagner's model of classical conditioning (Rescorla & Wagner 1972) assumes that learning occurs on a conditioning trial only if the US is surprising. "Surprise" is defined in terms of growth of "associative strength", the strength of the CS's association with the US over trials (V , traditionally measured in terms of number of URs). With each trial there is an increase or jump in associative strength. On early conditioning trials the jumps are large; that is, each trial causes a relatively large increase in associative strength. But the jumps decrease in size as learning progresses until the learning curve approaches its upper limit or asymptote. Once the CS predicts the US, the US is not surprising, and no further learning occurs.

Formally, for a CS s the change in learning on trial n is defined as

$$\Delta V_n(s) = \alpha\beta(\lambda - V_{n-1}(total)) \quad (1)$$

where α and β represent the salience of the CS and of the US respectively ($0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$), λ is the maximum amount of learning that can occur in that situation at that given trial, and $V_{n-1}(total)$ the cumulative amount of learning up to the previous trial, that is, the sum of associative strengths of all CSs that are present at trial n ; in turn, the associative strength of each of the CSs that are present is determined on the last trial on which each CS occurred, ordinarily trial $n - 1$. This delta rule is also known as the *error correction rule*: it calculates the prediction error, that is, the difference between the prediction and the actual reward. The result is then used to calculate the new associative strength of the CS as $V_n = V_{n-1} + \Delta V_n$, the *update rule*. Obviously, as the prediction improves the difference in delta is reduced until there is nothing left to be learned.

This deceptively simple theory is nevertheless considered as the most influential model of conditioning. Interestingly, Rescorla and Wagner's rule works pretty much as the Lagrangians in mechanics: during learning we balance what we have learned against what is to be learned so that the total associative strength is at each trial 1 (i.e., it is conserved) and the differences between trials, 0. In terms of optimization, Rescorla and Wagner's model uses equation (1) as a way of minimizing the prediction error between the expected reward and the actual reward – in other words, we apply an optimization principle that maximizes the reward.

Nonetheless, like most models of conditioning (see (Alonso & Schmajuk 2012) for a recent review) Rescorla and Wagner's is limited to classical conditioning: responses are only considered as a way of measuring how animals learn to associate two stimuli but do not form part of conditioning *per se*.

What happens when we study operant (aka instrumental) conditioning and goal-directed behaviour? In other words, what happens when the occurrence of a reinforcer depends on the choices the animal makes? Classical conditioning focuses on how “mental” representations of stimuli are linked whereas operant conditioning deals (mainly) with response-outcome associations. It is agreed though that, at the most general level, their *associative structures* are the same: in both procedures, changes in behavior are considered the result of an association between two concurrent events and explained in terms of operations of a (conceptual) system that consists of nodes among which links can be formed. Notwithstanding the correctness of such analysis, we are showing in the next section that a mere translation of classical conditioning into instrumental terms (for instance, by assuming that instrumental responses take the place of CSs) would impose a series of conditions on optimization that are impossible to meet. To see this point and understand our proposal to “recover” variational principles in the study of learning and behaviour we need to briefly introduce temporal difference, a model that comprises both classical and operant conditioning.

Temporal Difference

Temporal difference (TD) was originally presented as an extension of the Rescorla and Wagner's model in real time (Sutton & Barto 1987). It was argued that time scale invariance over trials should not prevent a model of conditioning from investigating temporal phenomena. Indeed, Rescorla and Wagner's model refers to learning through trials, and thus a number of interesting phenomena are left unexplained (such as second order conditioning). TD adopts Rescorla and Wagner's main psychological premises, namely, cue competition and error correction, but instead of comparing the rewards predicted on consecutive trials, we calculate the change in reward prediction error on every time step t . TD makes predictions over predictions and uses the error to update the old reward prediction and bring it more in line with the animal's moment-to-moment experiences – what is called *bootstrapping*.

Formally, in the general case the value of a CS at a particular time t is defined as

$$V_t(s_t) = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \gamma^3 r_{t+4} + \dots \quad (2)$$

where the γ parameter takes values between 0 and 1 and acts as a discount factor that causes distant CSs to matter less than immediate ones and r represents the US. If we compare the values at successive steps an interesting relationship emerges, namely,

$$V_t(s_t) = r_{t+1} + \gamma V_t(s_{t+1}) \quad (3)$$

This makes sense because $\gamma V_t(s_{t+1})$ takes the place of the remaining terms $\gamma r_{t+2} + \gamma^2 r_{t+3} + \dots + \gamma^{T-t-1} r_T$, where T refers to the terminal state, i.e., to the end of the trial. This relation describes the simplest TD case, when predictions are carried out one-step ahead. We can generalize it to any number of steps and calculate the delta rule as

$$\Delta V_t(s_t) = \alpha (R_t^{(n)} - V_t(s_t)) \quad (4)$$

where

$$R_t^{(n)} = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots + \gamma^{n-1} r_{t+n} + \gamma^n V_t(s_{t+n}) \quad (5)$$

If we take a number of steps into the calculation we would need to know how much each step contributes towards the return. TD proposes to average the n -steps with a trace λ (not to be mistaken for the US asymptotic value) so that the return is defined by

$$R_t^\lambda = (1 - \lambda) \sum_{n=1}^{T-t-1} \lambda^{n-1} R_t^{(n)} + \lambda^{T-t-1} R_t \quad (6)$$

And the new delta rule is

$$\Delta V_t(s_t) = \alpha (R_t^\lambda - V_t(s_t)) \quad (7)$$

It is easy to see that if λ is set to 0, TD(0), we only use the one-step backup; on the other hand, if λ is set to 1, TD(1), we only learn from the final return like in Rescorla and Wagner's model.

Temporal difference has gained notoriety because there is a strong correlation between its error term and the behaviour of dopamine cells in the brain (Montague, Dayan & Sejnowski 1996, Schultz 2002). Besides, TD focuses unashamedly on optimization and it is unique in that it aims at explaining both classical and instrumental conditioning. In fact TD has become the most successful Reinforcement Learning algorithm, bringing gaps between psychology, neuroscience, machine learning and control, and the new area of neuroeconomics (Glimcher, Camerer, Fehr & Poldrack 2009).

Temporal Difference and Operant Conditioning

Unlike Rescorla and Wagner's model, TD does provide a way (indirect as it might be) of learning how to select actions. The most common idea is to learn a separate value for each action leading out of a state, that is, executed in the presence of a stimulus, rather than for the state (stimulus) itself. An animal is assumed to exist in an environment described by some set of possible states S , where it can perform any actions A . Each time it performs an action a_t in some state s_t , the world enters into a new state $s_{t+1} = f(s_t, a_t)$ and the animal receives a real-valued reward $r_t = r(s_t, a_t)$. With this information, the animal calculates the TD error, typically using the so-called Q-learning rule (Watkins 1989), $\Delta V_t(s_t, a_t) = r_{t+1} + \gamma \max_{a_{t+1}} V_t(s_{t+1}, a_{t+1}) - V_t(s_t, a_t)$, and updates the value of the state-action pair as $V_{t+1}(s_t, a_t) = V_t(s_t, a_t) + \alpha \Delta V_t(s_t, a_t)$. The animal's task is to learn a control policy π , which maximizes the expected sum of rewards.

Under certain conditions Q-learning can be proved to converge to the value function that will yield the optimal policy. Tragically, Q-learning diverges when the state space is too large as it is the case in most biologically relevant problems.

A standard approach to tackle this problem is to introduce a scalar function approximator, $\tilde{V}(s, \vec{w})$ (e.g., a neural network with single output and weight vector). This is called the approximate value function, or the *critic*. The objective of learning is to make this function accurately estimate V^π for all S . We can then define a greedy policy on \tilde{V} as a policy that always considers all possible actions available to it and chooses the one that leads to the state with the highest \tilde{V} value, whilst also taking into account the immediate short terms reward in

getting there. The idea is to maximize the cumulative reward by minimizing the error as given by

$$\Delta \vec{w} = \alpha \sum \left(\frac{\partial \tilde{V}_t(s_t, a_t)}{\partial \vec{w}} \right) (R^\lambda - \tilde{V}_t(s_t, a_t)) \quad (8)$$

TD(λ) and Q-learning can then be used to update \tilde{V} by sampling one trajectory at a time. Variants of these methods have produced some successes in control problems (Tesauro 1994), yet TD algorithms have not been proved to converge in the general case. Why is that?

TD is based on Bellman's Optimality Condition (Bellman 1957): if $\tilde{V} \equiv V^\pi$ for all s in the state space S , where the policy is greedy on \tilde{V} , then that greedy policy is globally optimal. The problem is that for the Bellman's condition to be met we need to explore the entire state space. Even if Bellman's condition is perfectly satisfied along a single trajectory, performance can be extremely far from optimal if Bellman's condition is not satisfied over the neighbouring trajectories too. That is, if the animal tries to avoid Bellman's condition by only exploring a sub-space of the state space there is no guarantee the resulting policy will be locally optimal. This is the *curse of dimensionality* that applies to some degree to all value-based learning algorithms.

Translated into behavioural terms Bellman's condition means that for an animal to find an optimal policy it would need to explore all possible actions at every possible state. Clearly, this is not the way things happen in the natural world. To picture this, imagine Thorndike's cat trying to escape from the puzzle box. Bellman's condition would require that at every single step the cat would have to execute all the actions in its behavioural repertoire (including, for instance, banging its head against the wall). Hence, it is not just that Bellman's condition is computationally intractable. It is psychologically implausible too. Notice that in classical conditioning this problem does not arise since behaviours are reduced to reflexes. It is assumed that animals use a model (their evolutionary history) to "select" actions. In instrumental conditioning, however, any action can occur—at least in principle.

Value-Gradient Learning

In what follows we present a modification of TD, Value-Gradient Learning (VGL), that under certain conditions guarantees optimality. Importantly, VGL is equivalent to a variational principle, Pontryagin's Minimum Principle (PMP) (Pontryagin, Boltyanskii, Gamkrelidze & Mishchenko 1962) which, in turn, is a version of Hamilton's Principle of Least Action. The main difference between TD and VGL lies on *what* is learned: VGL learns gradients of values as opposed to TD algorithms that learn values. Besides, with regards to *how* learning occurs, VGL follows the gradient ascent on the total reward rather than the gradient descent on the expected reward.

We define the value gradient as

$$G(s) = \frac{\partial V(s, a)}{\partial s} \quad (9)$$

and the *approximate* value gradient as

$$\tilde{G}(s, \vec{w}) = \frac{\partial \tilde{V}(s, a)}{\partial s} \quad (10)$$

Our algorithm is defined by a weight update of the form

$$\Delta \vec{w} = \alpha \sum_t \left(\frac{\partial \tilde{G}}{\partial \vec{w}} \right)_t (G'_t - \tilde{G}_t) \quad (11)$$

where G'_t is the *target* value gradient defined recursively by

$$G'_t = \left(\frac{Dr}{Ds} \right)_t + \gamma \left(\frac{Df}{Ds} \right)_t (\lambda G'_{t+1} + (1 - \lambda) \tilde{G}_{t+1}) \quad (12)$$

with $G'_t = \vec{0}$ at any terminal state and where $\left(\frac{D}{Ds} \right)$ is shorthand for

$$\left(\frac{D}{Ds} \right) = \frac{\partial}{\partial s} + \frac{\partial \vec{a}}{\partial s} \frac{\partial}{\partial \vec{a}} \quad (13)$$

It has been proved that any greedy trajectory satisfying $\tilde{G}_t = G'_t$ for all t must be locally extremal, and often optimal (Fairbank, Prokhorov & Alonso 2012). This local optimality condition needs satisfying only over a single trajectory, whereas for TD the corresponding optimal condition (Bellman's) needs satisfying over the whole state space. It is easy to see that our demonstration is based on PMP. Unlike Bellman's condition, PMP states the *necessary* conditions for a trajectory to be (locally) optimal and thus it can be considered as a version of Bellman's Optimality Principle, if localized down to considering the current trajectory only. As a consequence, VGL can lead to increased efficiency. Moreover, it must be noticed that if we apply PMP to all the trajectories we "recover" global optimality.

Value-Gradient Learning and Temporal Difference

If we compare equation (11) against its TD equivalent (8), we see that they are analogous except for the introduction of the model in equation (12). More specifically, the definition of the target gradient G' is the full derivative with respect to s of the " λ -Return" which is the target used in the TD(λ) weight update. This may give the wrong impression that VGL(λ) is just a differentiated form of TD(λ). Contrarily, they differ in a fundamental way: in VGL, if the weight update is at a fixed point at every time step along a trajectory generated by a greedy policy, for any lambda, (i.e., if the learning objective $\tilde{G}_t = G'_t$ is met

for all t along the trajectory), then that trajectory is locally extremal, and often locally optimal (Fairbank *et al.* 2012). This contrasts to TD methods in that it is possible for the TD weight update to be at a fixed point at every time step along a trajectory generated by a greedy policy, without the trajectory being optimal. This is because for Bellman's condition to apply, the TD weight updates' objective needs satisfying over all of weight space, and hence lots of stochastic exploration is needed. Contrarily, VGL methods have a much lesser requirement for exploration. What we mean by this is that provided the VGL learning algorithm makes progress towards achieving $\tilde{G}_t = G'_t$ all along a greedy trajectory, then provided the trajectory remains greedy, it will make progress in *bending* itself towards a locally optimal shape, and this will happen without the need for any stochastic exploration. In comparison to VGL, the failure of TD without any exploration in a deterministic environment is dramatic and common, even when the value function is perfectly learned along a single trajectory.

The main insight is that it is not enough to use the derivatives of the values. This is what the Jacobi-Hamiltonian-Bellman equations do in extending the Bellman condition to continuous state spaces. Unfortunately, such derivation does not exploit fully the information contained in gradient values. We can't just consider the change in V over the particular step s along the trajectory. This is like "dotting" $\frac{\partial V}{\partial s}$ with Δs , which is approximately equal to the TD error in equation (4), once you add in r and include a discount factor.

In VGL, it is the *sideways* components of $\frac{\partial V}{\partial s}$ that are important, those that are *not parallel* to Δs . Such components are used in the calculation of G' , in the terms $\frac{Df}{Ds}$ and $\frac{Dr}{Ds}$ in particular. That is, you have to know these terms, that constitute the model function, in order to calculate a target value gradient, and you need a target in order to do a weight update.

In addition, the model function is relevant to the greedy policy. Using a first order expansion of the greedy policy gives

$$\pi(s, \vec{w}) = \arg \max_a \left(r(s, a) + \tilde{V}(f(s, a), \vec{w}) \right) \quad (14)$$

$$\approx \arg \max_a \left(r(s, a) + \tilde{V}(s, \vec{w}) + \left(\frac{\partial \tilde{V}(s, \vec{w})}{\partial s} \right)^T (f(s, a) - s) \right)$$

$$\approx \arg \max_a \left(r(s, a) + \left(\frac{\partial \tilde{V}(s, \vec{w})}{\partial s} \right)^T (f(s, a) - s) \right)$$

Hence the greedy policy depends on the value gradient but not on the values themselves. This is critical since changing $\frac{\partial \tilde{V}}{\partial s}$ will immediately affect the greedy policy; by

moving it towards its correct target we will steer the trajectory in the correct (locally optimal) direction: TD's paradigm "exploration vs. exploitation" becomes "exploration *and* exploitation" or, in other words, exploration comes for free when we combine "greedy" and "gradient" in VGL.

VGL is an extension of well-known methods in adaptive dynamic programming, Dual Heuristic Programming and Generalized Dual Heuristic Programming in particular, that have been proved to be successful in solving complex tasks such as autopilot landing, power system control, simple control benchmark problems such as "pole balancing", and many others (Wang, Zhang & Liu 2009). From a psychological point of view, VGL(1) is equivalent to Rescorla and Wagner's model. However, where as Rescorla and Wagner's model only considers classical conditioning VGL works for instrumental tasks –VGL, so to speak, is Rescorla and Wagner's model applied to operant conditioning.

Value-Gradient Learning and Pontryagin's Minimum Principle

In the next section we state how Hamilton's principle (aka the Principle of Least Action) and VGL apply to learning and behaviour. But first we need to be more precise about the relationship between VGL and Pontryagin's Minimum Principle.

As defined by Pontryagin, the Hamiltonian of a control system is a function of four variables: $\mathcal{H}(s, p, a, t) = \mathcal{L}(s, a, t) + p_t^T f(s, a, t)$ where $p_t = -\frac{\partial \mathcal{H}}{\partial s}$ is a costate interpreted as a Lagrange multiplier: If the state given by the function represents constraints in the minimization problem, the costate represents the cost of violating those constraints. In other words, p is the rate of change of the Hamiltonian as a function of the constraint. For example, in Lagrangian mechanics, the force on a particle $F = -\nabla V$ can be interpreted as p determining the change in action (transfer of potential to kinetic) following a variation in the particle's constrained trajectory. In economics, the optimal profit is calculated according to a constrained space of actions, where p is the increase in the value of the objective function due to the relaxation of a given constraint –the marginal cost of a constraint, called the shadow price.

Intuitively, the constraint f can be thought of as competing with the desired function to pull the system to its minimum or maximum (or to a steady state). And the Lagrange multiplier p can be thought of as measure of how hard f has to pull in order to make those forces balance out in the constraint surface.

Pontryagin's Minimum Principle (PMP) states that $\mathcal{H}(s_t^*, a_t^*, p_t^*, t) \leq \mathcal{H}(s_t, a_t, p_t, t)$ with the associated conditions for a maximum, namely, $p_t = -\frac{\partial \mathcal{H}}{\partial s}$, $s_t = \frac{\partial \mathcal{H}}{\partial p}$, and $\frac{\partial \mathcal{H}}{\partial a} = 0$. How is this related to VGL? Taking $\mathcal{H} = \mathcal{L} + pf$, we can make it correspond to VGL as follows: \mathcal{L} is the quantity to be maximized (or minimized), that is, the cumulative reward; the constraints are defined following the model of the world, f and r (henceforth, f for short); and p is G' (obviously \tilde{G} if the

trajectory is optimal, that is if $G' = \tilde{G}$). Hence we can express VGL in Hamiltonian form as $\mathcal{H} = r + G'f$. In fact, our re-formulation of PMP is somehow simpler, since PMP's conditions are reduced to two, namely, the costate and the max function that defines the greedy policy. At the end of the day, PMP can be described as $a^*(s, p) = \arg \min_a \mathcal{H}(s, p, a)$, which is a form of the greedy policy, and the adjoint equation $p_t \cong \frac{\partial V}{\partial s}(s_t, t)^T \in \mathbb{R}^n$, our gradient.

Let's recapitulate and see what happens with traditional value-based approaches: if there is no model, the Hamiltonian will not be constrained, thus it will be left to try all possible actions, not just those which "follow" the constraints. Indeed: without f , $\mathcal{H} = r + V'f$ reduces to $\mathcal{H} = r + V'$ –the old V formula.

Value-Gradient Learning and Behaviour Systems

To summarize, we have restored optimality. If we learn the gradient of the value function by choosing greedy actions that follow the full model of the system, Pontryagin's Minimum Principle applies and the trajectory so built is guaranteed to be locally optimal, that is, to minimize the error and to maximize the reward. This analysis begs the question: How does VGL apply to the study of behaviour?

At the end of the day, animals are behaviour *systems* – sets of behaviours that are organized around biological functions and goals like feeding (Timberlake 1983), defence (Fanselow 1994) or sex (Domjan 1994). When such systems are free to act as they please, their preferred or optimal distribution of activities defines a behavioural bliss point (BBP) or baseline level of activity. In dynamic terms the BBP is a natural, steady and stable, attractor.

This view encapsulates the behavioural regulation theory and generalizes the concept of homeostasis and negative feedback from physiology to psychology. Physiological homeostasis keeps parameters such as body temperature close to an optimal or ideal level. This level is "defended" in that deviations from the target temperature trigger compensatory physiological mechanisms that return the system to its homeostatic levels. In behavioural systems, what is defended is the organism's BBP against instrumental contingencies that create disturbances to which the system adapts. Other metaphors are possible: At the end of the day, the bliss point represents an equilibrium in a population of behaviours –pretty much as the equilibrium observed in the number of different types of ants in a colony or between competing (prey-predator) species in an environment.

More specifically, Staddon's model (Staddon 1979) explains operant behaviour in terms of time constraints and feedback constraints, the reinforcement schedule to which the animal is subjected. Starting from a BBP, the animal finds the optimal equilibrium between instrumental and contingent responses –the one that minimizes the cost involved. Instrumental conditioning procedures are seen as response constraints that disrupt the free choice of behaviour and prevent the organism from returning to the BBP. The organisms achieve a contingent optimization by

approaching its bliss point under the constraints of the instrumental conditioning procedure. Put it this way, the analysis of operant behaviour is an optimal control problem and thus we should be able to express it in terms of VGL: \mathcal{L} , the Lagrangian, is defined as the cost to be minimized, f are the time and feedback constraints and p , the multiplier or conjugate momentum, is now explicitly represented as G' . Not surprisingly, this formulation matches Staddon's term by term (see Appendix A, Staddon 1979).

Let's recapitulate, VGL's G value would be the gradient of the cost associated with a departure from a given distribution of actions. If the cost of a given distribution is represented as V , then $G(s) = \frac{\partial V(s,a)}{\partial s}$ represents the change in cost as we change the distribution –where s represents the distribution and a represents a given set of responses (both instrumental and contingent). G' re-acts against the constraints to minimize the cost.

What are the advantages of using VGL? Firstly, VGL tells us exactly which form the multiplier must have. In particular, G' must be defined according to $\frac{Dr}{Ds}$ and $\frac{Df}{Ds}$: the former tells us how the rate of contingent responses (r) changes as the distribution of responses changes and the latter how the constraints themselves change. These two quantities define the change of cost that we minimize and give us the optimal distribution.

Perhaps more importantly, VGL does not only give a solution to an optimization problem –in this case, the optimal distribution of responses under certain constraints. Of course, it does if we assume that such functions are perfectly known; yet, VGL is also a learning algorithm and as such serves a mechanistic agenda as well as an equilibrium agenda. VGL allows us to calculate how the animal is adapting to the optimal distribution when the constraints are a *moving* target, solving the so-called “teleological conundrum”: of course, animals do not know what the reinforcement schedule would be or the corresponding optimal response ratio –and yet they adapt to the optimal solution and they do so in an optimal way. Perhaps an analogy may clarify this point: Physicists found it puzzling that particles behaved as if they knew what the future would be. Traditionally, the movement of particles was interpreted in terms of global symmetries and thus it was difficult to explain how particles abided by the Principle of Least Action locally, when constraints appeared and disappeared as the system interacted with “unexpected” forces. Surely, the symmetries were broken in such cases; and yet, Nature seemed to account for them so as to comply with global symmetries –“as if nothing had happened”, symmetry was restored. We know that the answer lies in *gauge* symmetries: Indeed, at each step, deviations are counter-balanced so as to bring the system back (or as close as possible) to the original symmetry. In terms of cognition, this is precisely what VGL does.

Conclusion

This paper does not present quantitative predictions or new results. It presents a formal model that integrates current theories of conditioning with fundamental principles of Nature. Our main assumption is that learning

and behaviour, conditioning more in particular, follow the same variational principles as any other natural phenomena: they must make a functional of some sort of extremal. In that we follow Peter Killeen's program (Killeen 1992). We have shown that Temporal Difference is an inadequate model of optimal behaviour and proposed a new model, Value-Gradient Learning, equivalent to Pontryagin's Minimum Principle –in turn, a version of Hamilton's Principle of Least Action, that may serve as a model of both classical and operant conditioning.

References

- Alonso, E., & Schmajuk, N. (2012). (Eds.), Special Issue on Computational Models of Classical Conditioning, *Learning & Behavior*, No.: 3.
- Bellman, R. (1957). *Dynamic Programming*. Princeton, NJ: Princeton University Press.
- Domjan, M. (1994). Formulation of a behavior system for sexual conditioning. *Psychonomic Bulletin & Review*, 1, 421-428.
- Fairbank, M., Prokhorov, D., & Alonso, E. (2012). Approximating Optimal Control with Value Gradient Learning. In J. Si, A.G. Barto, W.B. Powell & D. Wunsch (Eds.), *Handbook of Learning and Approximate Dynamic Programming, Vol. 2*. Wiley-IEEE Press.
- Fanselow, M. S. (1994). Neural organization of the defensive behavior system responsible for fear. *Psychonomic Bulletin & Review*, 1, 429-438.
- Glimcher, P.W., Camerer, C.F., Fehr, E., & Poldrack, R.A. (2009). *Neuroeconomics: Decision Making and the Brain*. San Diego, CA: Academic Press.
- Killeen, P.R. (1992). Mechanics of the animate. *Journal of the experimental Analysis of Behavior*, 57, 429-463.
- Montague, P. R., Dayan, P., & Sejnowski, T. J. (1996). A framework for mesencephalic dopamine systems based on predictive Hebbian learning. *Journal of Neuroscience*, 16, 1936-1947.
- Pontryagin, L.S., Boltyanskii, V.G., Gamkrelidze, R.V., & Mishchenko, E.F. (1962). *The Mathematical Theory of Optimal Processes*. New York, NJ: Gordon and Breach Science Publishers.
- Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement. In A. H. Black & W. F. Prokasy (Eds.), *Classical conditioning: II. Theory and research* (pp. 64-99). New York: Appleton-Century-Crofts.
- Schultz, W. (2002). Getting formal with dopamine and reward. *Neuron*, 36, 241-263.
- Staddon, J.E. (1979). Operant behavior as adaptation to constraint. *Journal of Experimental Psychology: General*, 108, 48-67.
- Sutton, R. S., & Barto, A. G. (1987). A temporal difference model of classical conditioning. In *Proceedings of the Seventh Annual Conference of the Cognitive Science Society* (pp. 355-378). Erlbaum.
- Tesauro, G.J. (1994). TD-Gammon, a self-teaching backgammon program, achieves master level play. *Neural Computation*, 6, 215-219.
- Timberlake, W. (1983). Rats' responses to a moving object related to food or water: A behavior-systems analysis. *Animal Learning & Behavior*, 11, 309-320.
- Wang, F.-Y., Zhang, H., & Liu, D. (2009). Adaptive dynamic programming: An introduction, *IEEE Computational Intelligence Magazine*, May, 39-47.
- Watkins, C.J.C.H. (1989). *Learning from Delayed Rewards*. Ph.D. Thesis, Cambridge University.

Annual Cognitive Modeling Competition

Kevin Gluck (kevin.gluck@wpafb.af.mil)
Air Force Research Laboratory
711 HPW/RHAC
2620 Q Street, Bldg 852
Wright-Patterson AFB, OH 45433-7955 USA

Keywords: Cognitive; Modeling; Competition

Opportunity

Although the cognitive and behavioral modeling communities now have a rather lengthy history and there is ongoing research and development in these areas across dozens of academic, industrial, and government research laboratories, very little of the work has been explicitly competitive in orientation. That said, there are precedents for modeling competitions in this area. Two examples include the PokerBot Competition (Lebiere & Bothell, 2004) and the Dynamic Stocks and Flows Model Comparison Challenge (Lebiere, Gonzalez, & Warwick, 2010). Both of these were successful and interesting events. However, they were also both single shot modeling competitions that did not evolve into annual events in the spirit of the Robocup (2012) robotic soccer competitions.

I propose it is time to establish a recurring annual cognitive modeling competition. This poster is an opportunity for community discussion of the pros, cons, infrastructure requirements, and design parameters that should be considered in developing such an event within the cognitive and behavioral sciences.

Acknowledgments

Many thanks to the Air Force Research Laboratory's Office of Scientific Research and Human Effectiveness Directorate for the resources necessary to attend the conference and dialogue with the research community regarding an annual cognitive modeling competition.

References

- Lebiere, C., Gonzalez, C., & Warwick, W. (2010). Cognitive architectures, model comparison, and AGI. *Journal of Artificial General Intelligence*, 2, 1-19. (see also <http://www.hss.cmu.edu/departments/sds/ddmlab/models/index.html>)
- Lebiere, C., & Bothell, D. (2004). Competitive modeling symposium: Pokerbot World Series. In *Proceedings of the 2004 International Conference on Cognitive Modeling*. p.32. Mahwah, NJ: Erlbaum. (see also <http://www.lrdc.pitt.edu/schunn/ICCM2004/index.htm>)
- RoboCup. (2012). RoboCup. Retrieved from the web January 6, 2012. <http://www.robocup.org/>

Choices, Choices: Task Selection Preference During Concurrent Multitasking

Menno Nijboer (m.nijboer@rug.nl)

Department of Artificial Intelligence, Nijenborgh 9
Groningen, 9747 AG, NL

Niels Taatgen (n.a.taatgen@rug.nl)

Department of Artificial Intelligence, Nijenborgh 9
Groningen, 9747 AG, NL

Hedderik van Rijn (hedderik@van-rijn.org)

Department of Experimental Psychology, Grote Kruisstraat 2 /1
Groningen, 9712 TS, NL

Keywords: Threaded cognition; multitasking; interference; cognitive models

Introduction

Multitasking

With the ever-increasing stream of information we are expected to deal with on a moment-to-moment basis, human multitasking behavior has become an important part of modern society.

Multitasking can occur on many different timescales. Our interest is in concurrent multitasking: attempting to fulfill multiple goals in parallel. There have been many investigations to determine whether concurrent multitasking is good or bad. However, there is no definite answer to this question. Instead, it seems to depend very much on the tasks that are performed concurrently, as well as the amount of experience one has with the tasks. For instance, studies into driving behavior have shown that purely cognitive tasks can have a negative impact on driving performance (Horrey & Wickens, 2006). On the other hand, some studies have shown that perfect multitasking is possible (Schumacher et al., 2001).

Early attempts to explain the results of multitasking studies revolved around multiple resource theories. These postulate that the cognitive system can be divided up into several resources. Once the capacity of a resource is exceeded, it can create interference during multitasking. While able to offer explanations for multitasking observations, these theories cannot produce detailed models that can be used to predict behavior in new situations. However, with the development of cognitive architectures, our ability to predict multitasking behavior has greatly increased.

Threaded Cognition

In efforts to explain concurrent multitasking behavior, threaded cognition has shown to be a very effective theory. Threaded cognition was developed by Salvucci and Taatgen (2008), and is implemented in the ACT-R cognitive architecture (Anderson, 2007). As such, it follows the

constraints imposed by ACT-R: the cognitive system can be divided into different resources (such as vision, working memory, and manual control) that can operate in parallel. Each resource can only be used by one task at any given time, however.

In threaded cognition, multiple goals can be active at the same time. As such, explicit goal switching is no longer required. Furthermore, allocation of the resources is based on two principles: politeness and greediness. Greediness means that a task can use a resource if that resource is not in use by another task. Politeness states that a task will immediately release a resource when it is done using that resource.

Threaded cognition has been successful in explaining a wide range of multitasking behavior, such as multitasking in driving, track and choice experiments (Salvucci & Taatgen, 2008), and perfect time-sharing experiments (Schumacher et al., 2001).

Task Selection

While threaded cognition has helped us in our understanding of multitasking, it has not yet explained how people determine which task to perform. Motivation is considered to play a large role in selecting and executing goals (Vancouver et al., 2010). However, we believe that cognitive factors also play an important role: interference that arises between two tasks that require the same resource at the same time leads to reduced performance and increased execution times. Intuitively, this is something that people will try to avoid. As such, cognitive factors can affect which tasks people will prefer to perform concurrently.

Our hypothesis is that when people have to choose between combinations of tasks, they will choose the combination that has the smallest resource conflict.

Study

To examine the effect of cognitive interference on task selection, we performed a study involving concurrent multitasking.

Methodology

20 participants (13 female, mean age 22.2) performed a dual task experiment consisting of a math task combined with either an aural/declarative task or a visual/manual task.

The math task was a 10-column subtraction sum that had to be solved in a right to left order. The task had an easy and a hard version. In the hard version participants had to remember if they borrowed at the previous column. In the easy version, no borrowing was required.

The visual/manual task was a tracking task (Martin-Emmerson & Wickens, 1992). Using a trackball, participants had to push a moving dot back into a circle. Each time the dot went outside the circle, an error buzzer would sound.

The aural/declarative task was a tone counting task. During a trial, tones would be presented to participants through a pair of headphones. After completing the last digit of the subtraction task, participants were prompted to type in the number of tones they heard.

The study consisted of a practice block and two main blocks A and B. Block A consisted of trials where the subtraction difficulty and task combination was fixed. Participants performed 4 trials of each combination. In block B only the subtraction difficulty was fixed. Before each trial the participants could choose whether they wanted to perform tone counting or tracking.

In block B, when subtraction is easy, we expect that subjects will choose tone counting most often, because there is no resource overlap between those two tasks, while the tracking tasks shares both visual and manual resources. However, when the subtraction task is hard, there is interference in the problem state (working memory) resource, making it more likely that subjects will choose tracking.

Results

An analysis of the block B data shows that participants almost exclusively choose tone counting when the subtraction task is easy. When faced with a difficult subtraction problem, there is a shift towards choosing tracking instead of tone counting: when subtraction is easy counting tones is chosen in 93% of the trials. When subtraction is hard, tone counting is chosen in only 73% of the trials ($p < 0.05$, $df = 38$, $F = 5.0$).

Further examination of the performance on the tone counting and tracking tasks shows that while tracking performance does not change depending on the subtraction difficulty, participants are substantially worse in tone counting when the subtraction task is hard: 78% vs. 46% correct ($p < 0.01$, $df = 38$, $F = 9.23$).

Conclusion

Our main interest lies in the selection of the secondary task in block B. Given threaded cognition and the constraints imposed by ACT-R, our hypothesis is that when presented with an easy subtraction, participants will choose the tone

counting task as there is minimal overlap in the resources used by both tasks. In the hard subtraction condition, however, we expect participants to choose for the tracking task. Even though tracking requires participants to look away from the subtraction task, it does not result in interference that might arise from remembering both the tone count and a possible borrow performed in the previous subtraction column.

The results support our hypothesis: participants almost exclusively pick tone counting in the easy subtraction condition, but are more likely to choose for tracking in the hard subtraction condition, despite the more distracting nature of the tracking task.

Interestingly, participants still pick tone counting in combination with hard subtraction, despite making more errors in counting. This suggests that feedback might play an important role in task selection: the tracking task provides continuous feedback during the trial, while the tone counting task only has one feedback moment at the end of the trial. Furthermore, this feedback has no real consequence for the participant when an incorrect answer was given. In contrast, the tracking task produces an error buzzer when the dot is no longer in the circle. This lack of negative feedback seems to make participants less sensitive to poor performance in the counting task, which could explain the preference for this task in the hard subtraction condition. This hypothesis will be tested in a follow-up study.

Acknowledgements

This research was funded by ERC-StG grant 283597 awarded to Niels Taatgen.

References

- Anderson, J.R. (2007). How can the human mind exist in the physical universe? OUP, New York.
- Horrey, E., Wickens, C. D., (2006). Modeling drivers' visual attention allocation while interacting with in-vehicle technologies. *Human Factors* 48 (196).
- Martin-Emerson, R., Wickens, C. D., (1992). In *Proc. Human Factors*. Human Factors Society, Santa Monica, CA, 1408-1412
- Martin-Emerson, R., Wickens, C. D., (1992). The vertical visual field and implications for the head-up display. In *Proc. Human Factors*. Human Factors Society, Santa Monica, CA, 1408-1412.
- Salvucci, D. D., & Taatgen, N. A. (2007). Threaded Cognition: An Integrated Theory of Concurrent Multitasking. *Psych. Rev.*, 115, 101-130.
- Schumacher, E. H., Seymour, T. L., Glass, J. M., Fencsik, D. E., Lauber, D. E., Kieras, D. E., & Meyer, D. E. (2001). Virtually Perfect Time Sharing in Dual-Task Performance: Uncorking the Central Cognitive Bottleneck. *Psych. Sci.*, 12, 101-108
- Vancouver, J. B., Weinhardt, J. M., & Schmidt, A. M., (2010). A formal, computational theory of multiple-goal pursuit: integrating goal-choice and goal-striving processes. *Journal of Applied Psych.*, 95 (6), 985-1008

Extending and Evaluating a Multiplicative Model for Semantic Composition in a Distributional Semantic Model

Akira Utsumi (utsumi@inf.uec.ac.jp)

Department of Informatics, The University of Electro-Communications
1-5-1, Chofugaoka, Chofushi, Tokyo 182-8585, Japan

Abstract

This paper addresses a multiplicative model for semantic composition in a distributional semantic model. This paper proposes new composition algorithms using a multiplicative model by two approaches: to extend a multiplicative model by averaging or weighting and to modify context-sensitive additive models such as Kintsch's (2001) predication by replacing vector addition with vector multiplication. In addition, this paper examines conditions for the superiority of the multiplicative models found in previous research by comparing two semantic spaces constructed by latent semantic analysis (LSA) and positive pointwise mutual information (PPMI) in terms of the representational ability of composition algorithms. The experiment using noun compounds demonstrated that the multiplicative model performed better than the additive model only in the PPMI-based space, suggesting that component-wise multiplication works effectively in a semantic space whose dimensions represent distinctive features. Some multiplicative modifications of additive algorithms also improved the performance in the PPMI-based space, but did not in the LSA-based space. Interestingly, however, the extension of the multiplicative model by weighting was effective in improvement of the performance in the LSA-based spaces, although the extensions of the multiplicative model were not effective in the PPMI-based space.

Keywords: Distributional semantic models; Semantic spaces; Vector composition; Component-wise multiplication; Latent semantic analysis; Pointwise mutual information

Introduction

Recently there is a growing interest in vector-based semantic space models or distributional semantic models in the field of cognitive science (e.g., Landauer, McNamara, Dennis, & Kintsch, 2007; McNamara, 2011) as well as in natural language processing (e.g., Padó & Lapata, 2007; Turney & Pantel, 2010). The primary reason for the growing interest is that despite its simplicity the semantic space model has demonstrated high performance of cognitive modeling for a number of cognitive tasks such as similarity judgment (Landauer & Dumais, 1997), semantic priming (Jones, Kintsch, & Mewhort, 2006), and metaphor comprehension (Utsumi, 2011). In particular, the ability of the semantic space model to represent the meanings of single words has been extensively examined in a number of studies. However, less attention has been paid to the problem of how to construct the semantic representations of larger linguistic units such as phrases and sentences. Given the creativity of human language that an infinite number of phrases or sentences can be constructed from a finite number of words, distributional semantic models should provide appropriate methods for constructing (or composing) vectors for phrases or sentences from the vectors of their constituent words. This paper addresses this problem.

In the previous studies on vector composition (e.g., Baroni & Zamparelli, 2010; Mitchell & Lapata, 2008, 2010; Zanzotto et al., 2010), two classes of composition methods,

namely *an additive model* and *a multiplicative model*, have been proposed and evaluated in a number of tasks. The widely-used algorithm in the additive class is to compute the centroid of constituent word vectors. Although the centroid method simply combines the contents of all the constituent words involved in the target phrase or sentence, more sophisticated algorithms in the additive class such as predication (Kintsch, 2001) and comparison (Utsumi, 2011) use additional information (i.e., words that are semantically similar or related to the constituent words) to compute contextually dependent meanings. These additive algorithms have been shown to be effective in a number of applications of a distributional semantic model (e.g., Kintsch, 2001; Landauer et al., 2007). However, recent studies (Mitchell & Lapata, 2008, 2010) have demonstrated that component-wise multiplication of multiple vectors, one representative method of the multiplicative class model, performs better than the centroid algorithm and other additive methods. This finding is particularly interesting because until recently the centroid algorithm and other additive models have been widely accepted as an effective method for vector composition. Hence, this paper is concerned with the multiplicative model and tackles two issues concerning the multiplicative model.

One issue addressed in this paper is why and when a multiplicative model performs better than an additive model. As mentioned above, some studies have found the superiority of the multiplicative model in general and component-wise multiplication in particular, but no studies have revealed the rationale behind, and conditions for, the superiority of the multiplicative model. Our working hypothesis is that, unless dimensions of a semantic space represent distinctive features or meanings, component-wise multiplication does not work effectively, because its function is to highlight dimensions relevant to both vectors and downplay irrelevant dimensions. This paper examines the validity of this hypothesis using (at least) two different types of semantic spaces whose dimensions represent or do not represent distinctive features.

Another important issue of interest is to propose new composition algorithms on the basis of the superiority of component-wise multiplication. Two approaches are considered to devise a method for semantic composition. One approach is that we can extend the component-wise multiplication algorithm by averaging or weighting. The purpose of this modification is to adjust the degree of highlighting relevant dimensions to a more appropriate level. Weighting is also aimed at involving the information of word order in a composition vector. Another approach is to modify the additive model using a multiplicative function. For example, the predication algorithm (Kintsch, 2001) generates the composition vector of a two-word phrase by computing the centroid of the both words and some semantic neighbors of a predicate word that are also related to an argument word. If the multiplica-

tive model generally performs better than the additive model, it naturally follows that we can obtain new and possibly efficient algorithms by replacing vector addition in the predication algorithm with component-wise multiplication. In this paper, we propose new algorithms for vector composition by adopting these approaches, and compare the representational ability of these algorithms with the original ones.

In order to evaluate the ability of new composition algorithms, we conducted an experiment using two-noun compounds. In the experiment, the vector for noun compounds was computed by the composition algorithms, and its representational ability was evaluated using the plausibility of the semantic relatedness or similarity computed between vectors for noun compounds and semantically related words. The plausibility of the similarity was evaluated by two measures: correlation between the computed cosine similarity and human similarity judgment for compound-word pairs, and ranking of the similarity of compound-word pairs. Furthermore, we also examined the hypothesis that multiplicative models perform differently depending on the types of semantic spaces using two different semantic spaces. The experiment was conducted in two languages, i.e., English and Japanese, to test whether different languages show consistent results.

Composition Algorithm

In this section, we review existing methods for vector composition and propose new methods by extending or modifying the existing methods. Throughout this section, we explain the methods by assuming that we are generating the vector v for a two-word phrase or sentence $P(A)$ that comprises the argument (or the head) A (whose vector is denoted by a) and the predicate (or the modifier) P (whose vector is denoted by p). For example, a vector for the sentence “A horse runs” is computed from the vector for the argument “horse” and the vector for the predicate “run.”

Reviewing the existing composition algorithms

The simplest but widely used additive method for vector composition is to compute the *centroid* of constituent word vectors i.e., $v = (p + a)/2$. Note that, when the cosine is used as a similarity measure, additive models yield the same result of similarity computation, whether vectors are averaged or simply summed. Hence, the simple addition (i.e., $v = a + p$) suffices as the centroid algorithm.

A more sophisticated method, namely *dilation*, in the additive class is proposed by Mitchell and Lapata (2010); this method stretches or “dilates” the argument (or head) vector along the direction of the predicate (or modifier) vector. This can be done by decomposing the argument vector a into two orthogonal vectors, i.e., a vector x parallel to p and a vector y orthogonal to p , and then stretching the parallel component x so that a modified vector v of a is more like p .

$$\begin{aligned} v &= \lambda x + y = \lambda \frac{p \cdot a}{p \cdot p} p + \left(a - \frac{p \cdot a}{p \cdot p} p \right) \\ &= (\lambda - 1) \frac{p \cdot a}{p \cdot p} p + a \end{aligned} \quad (1)$$

In Equation 1, λ is a parameter that represents the degree of dilation. When $\lambda = 1$, the resulting vector v is identical to the

argument vector a . Mitchell and Lapata (2010) demonstrated that the dilation algorithm performs consistently well on a phrase similarity task.

The *predication* algorithm proposed by Kintsch (2001) does not simply combine the constituent vectors, but embodies the semantic interaction between constituent words on the basis of his construction-integration model. The predication algorithm makes use of semantic neighbors (i.e., words similar to the constituent words) to embody the interaction; it first chooses m nearest neighbors of a predicate P (i.e., m words with the highest cosine to P) and then picks up k neighbors of P that are also related to A . Finally the algorithm computes the centroid vector of P , A , and the k neighbors of P as a vector representation of $P(A)$. Formally, the predication algorithm computes a composition vector v by

$$v = p + a + \sum_{q_i \in Q} q_i, \quad (2)$$

where Q denotes a set of the k neighbors of P that are also related to A .

Utsumi (2011) proposes a different algorithm for vector composition, namely, *comparison*, to compute the meanings of metaphors. This algorithm uses a set of common neighbors of two constituent words to capture the relevance between them. It is motivated by the fact that the interaction between constituent words embodied in the predication algorithm is rather predicate-directed. Formally, the comparison algorithm computes a compound vector p using the

$$v = a + \sum_{c_i \in C} c_i, \quad (3)$$

where C denotes a set of k common neighbors of A and P . The set C of common neighbors can be obtained by finding the smallest i that satisfies $|N_i(A) \cap N_i(P)| \geq k$, where $N_i(w_j)$ denotes a set of the top i neighbors of the word w_j .

Turning now to the multiplicative model, the simplest method is to use a function of *component-wise multiplication* for vector composition:

$$v = p \odot a \quad (4)$$

$$v_i = p_i \cdot a_i \quad (5)$$

where the symbol \odot expresses that two vectors are multiplied component-wise.

Circular convolution is also a member of the multiplicative class, and defined as

$$v = p \otimes a \quad (6)$$

$$v_i = \sum_{j=0}^{n-1} p_{j \bmod n} \cdot a_{(i-j) \bmod n}. \quad (7)$$

Proposing new composition algorithms

Extension of component-wise multiplication This approach extends the component-wise multiplication algorithm in Equation 4 in two ways: averaging and weighting.

$$v_i = \sqrt{p_i \cdot a_i} \quad (8)$$

$$v_i = p_i^\alpha \cdot a_i \quad (9)$$

In the averaging approach expressed in Equation 8, each component of a composed vector is calculated as a geometric mean, rather than a simple multiplication. This extended algorithm can be regarded as corresponding to the centroid algorithm in the additive class. On the other hand, in the weighting approach in Equation 9, the predicate vector is weighted by the factor α . When $0 < \alpha < 1$, the argument vector has a stronger influence on the resulting composition vector, while the predicate vector has when $\alpha > 1$. This modification can be regarded as a multiplicative version of the dilation algorithm.

Modification of context-sensitive additive models using a multiplicative model This approach modifies the context-sensitive additive models, namely the predication and comparison algorithms, to benefit from the multiplicative model.

One way of modifying these additive models is to replace the vector addition (+) with the vector multiplication (\odot). For example, the predication algorithm can be modified as

$$v = p \odot a \odot \prod_{q_i \in Q} q_i \quad (10)$$

$$v = \sqrt{(k+2)} p \odot a \odot \prod_{q_i \in Q} q_i. \quad (11)$$

Equation 10 shows a simple multiplicative version of the predication algorithm, in which all the argument, predicate, and neighbor vectors are combined by component-wise multiplication. On the other hand, Equation 11 is an averaged version of the multiplicative predication algorithm, in which the argument, predicate, and neighbor vectors are geometrically averaged. (Note that in these equations the product symbol denotes the product by component-wise multiplication, and the radical symbol denotes the component-wise root.)

Another way of modifying the additive models is to keep the sum of neighbor vectors unchanged, and multiply the argument, predicate, and the sum of neighbor vectors.

$$v = p \odot a \odot \sum_{q_i \in Q} q_i \quad (12)$$

$$v = \sqrt[3]{p \odot a \odot \sum_{q_i \in Q} q_i} \quad (13)$$

This modification is motivated by the assumption that the contextually dependent meaning of a predicate should be computed by the disjunction of neighbors rather than by the conjunction of neighbors. For example, according to this assumption, the meaning of the predicate *run* in “A horse runs” should be represented as “move, flee, **or** walk,” rather than “move, flee, **and** walk”

The comparison algorithm can be modified by the same approaches embodied in Equations 10–13. (These modifications are listed in Table 1.)

Method

Materials

Compound-word pairs Noun compounds we used in the experiment comprised two nouns (i.e., head and modifier)

such as “apple pie” or “ruling class.” These compounds included two types: *familiar noun compounds* that occur in the corpus from which semantic spaces were constructed and *novel noun compounds* that do not occur in the corpus. The selection criterion for compounds was that familiar compounds should occur at least 20 times in the corpus, and both types of compounds should be included in the thesaurus. From the compounds that satisfied this criterion, we randomly selected 50 familiar and 50 novel compounds in both languages for evaluation. For each of these noun compounds, a semantically related word was selected randomly from synonyms, hypernyms and coordinate words of that compound.

In this experiment, the written and non-fiction parts of the British National Corpus of the size of 54.7 million words and Japanese newspaper corpora (i.e., four years’ worth of Mainichi newspaper articles and two year’s worth of Nikkei newspaper articles) of the size of 26.2 million words were used as a corpus. They contained 73,422 English and 63,875 Japanese different words. The thesauri used in this experiment were English WordNet 3.0 and a Japanese thesaurus “Nihongo Dai-Thesaurus.”

Semantic space In order to test our working hypothesis that component-wise multiplication works effectively in the semantic space whose dimensions represent distinctive features, we used two different semantic spaces.

One semantic space is based on the word-word matrix whose elements are word cooccurrence frequencies within a context window spanning some number of words. This model is a popular semantic space (e.g., Bullinaria & Levy, 2007; Recchia & Jones, 2009) and it was also used by Mitchell and Lapata’s (2008, 2010) study demonstrating the superiority of the component-wise multiplication. Formally, the element a_{ij} of the word-word matrix is initially the number of times the word w_j occurs within n words around the word w_i , and weighted by positive pointwise mutual information (PPMI; Bullinaria & Levy, 2007; Turney & Pantel, 2010). In this study, we used a context window of five words, i.e., $n = 5$.

Another semantic space was constructed using latent semantic analysis (LSA; Landauer & Dumais, 1997; Landauer et al., 2007). LSA is based on the word-document matrix whose element a_{ij} is the number of times the word w_i occurs in the j -th document. The elements of this initial matrix are weighted, and the matrix is smoothed by singular value decomposition (SVD). Among a number of weighting schemes that have been proposed so far, we used Quesada’s (2007) scheme in which the initial word frequency is weighted by the product of its logarithm and the entropy. The number of the reduced dimensions was determined to be 300.

One essential difference between these two semantic spaces lies in the meaningfulness of vector dimensions. In the PPMI-based semantic space, each vector component represents a distinctive feature, namely, a context word, while the dimensions of the LSA-based semantic space do not have such the clear meaning. Hence, by comparing these two spaces in terms of the performance of the multiplicative models, we tested the working hypothesis that component-wise multiplication works well in a semantic space with semantically meaningful dimensions. In addition, to test whether the smoothing by SVD is a main cause of the loss of dimension semantics, we also examined the performance of the PPMI-

Table 1: Composition algorithms compared in the experiment

Algorithm (abbr.)	Function
Centroid (CENT)	$v_i = p_i + a_i$
Dilation (DILA)	$v_i = (\lambda - 1)p_i \sum_j p_j a_j + a_i \sum_j p_j p_j$
Predication (PRED)	$v_i = p_i + a_i + \sum_j q_{ji}$
Multiplicative (+MUL)	$v_i = p_i \cdot a_i \cdot \prod_j q_{ji}$
Geometrically averaged (+AVE)	$v_i = \sqrt{(k+2)} p_i \cdot a_i \cdot \prod_j q_{ji}$
Partially multiplicative (+PARTMUL)	$v_i = p_i \cdot a_i \cdot \sum_j q_{ji}$
Partially averaged (+PARTAVE)	$v_i = \sqrt[3]{p_i \cdot a_i \cdot \sum_j q_{ji}}$
Comparison (COMP)	$v_i = a_i + \sum_j c_{ji}$
Multiplicative (+MUL)	$v_i = a_i \cdot \prod_j c_{ji}$
Geometrically averaged (+AVE)	$v_i = \sqrt{(k+1)} a_i \cdot \prod_j c_{ji}$
Partially multiplicative (+PARTMUL)	$v_i = a_i \cdot \sum_j c_{ji}$
Partially averaged (+PARTAVE)	$v_i = \sqrt{a_i \cdot \sum_j c_{ji}}$
Multiplication (MULT)	$v_i = p_i \cdot a_i$
Averaged (+AVE)	$v_i = \sqrt{p_i \cdot a_i}$
Weighted (+WEI)	$v_i = p_i^\alpha \cdot a_i$
Convolution (CONV)	$v_i = \sum_{j=0}^{n-1} p_j \bmod n \cdot a_{(i-j) \bmod n}$
Head only (HEAD)	$v_i = a_i$
Vector (VECT)	v is computed directly from the corpus by treating compounds as single words

based semantic space smoothed by SVD.

Human similarity judgment

In order to collect the data on human similarity judgment, we conducted an experiment using Japanese compound-word pairs. (English compound-word pairs were not used for the experiment, because a sufficient number of native English speakers could not be recruited.) Fourteen participants, who were all native speakers of Japanese, were assigned all the 100 Japanese compound-word pairs and asked to rate the semantic relatedness between the compound and the word of each pair. These pairs were rated on a 7-point scale ranging from 1 (*unrelated*) to 7 (*related*). The presentation order of those pairs was randomized for each participant.

Procedure

Given a semantic space and a set P of compound-word pairs, the compound vector of each compound was computed by the composition algorithms. Afterward, the similarity between the compound and the paired word was calculated as the cosine between the computed compound vector and the vector for the paired word. Using these cosine values for the set of compound-word pairs, we evaluated the composition algorithms in the following two ways.

Correlation analysis Spearman’s correlation coefficient was calculated between the computed cosine values and the mean human ratings for compound-word pairs.

Word ranking For each compound-word pair (c_i, w_i) , the rank r_i of the paired word w_i was assessed by computing the cosine similarity between c_i and all words (including w_i) in the space, and sorting all words in descending order of the cosine. A higher rank implies that the word w_i is semantically more related to the compound c_i . Next, all compound-word

Table 2: Correlation coefficients between human similarity ratings and the cosine similarity computed by the composition algorithms

Algorithm	PPMI		LSA		PPMI+SVD	
	Fam	Nov	Fam	Nov	Fam	Nov
CENT	.376	.381	.367	.319	.446	.466
DILA	.359	.348	.443	.405	.445	.422
PRED	.410	.343	.378	.264	.433	.422
+MUL	.388	.446	.446	.310	.322	.094
+AVE	.331	.412	.474	.213	.408	.321
+PARTMUL	.340	.432	.446	.308	.346	.108
+PARTAVE	.336	.407	.441	.218	.403	.345
COMP	.360	.331	.346	.403	.432	.410
+MUL	.219	.397	.296	.413	.442	.127
+AVE	.247	.333	.325	.239	.461	.348
+PARTMUL	.245	.355	-.238	-.079	.057	.050
+PARTAVE	.262	.314	-.249	-.218	.051	.122
MULT	.393	.414	-.149	.005	.037	.002
+AVE	.369	.413	-.194	-.086	.000	.056
+WEI	.392	.409	.281	.437	.483	.310
CONV	.294	.123	.216	.347	.418	.386
HEAD	.287	.260	.361	.411	.383	.365
VECT	.331	—	.416	—	.404	—

■ $p < .05$, ■ $p < .01$, ■ $p < .001$

Fam=Familiar compounds, Nov=Novel compounds.

pairs in the set P were sorted in ascending order of the rank r_i . The sorted list of the rank r_i is denoted as $r'_1, \dots, r'_{|P|}$. Finally, the overall performance of each algorithm was measured by the median rank $\simeq r'_{0.5|P|}$ and first quartile rank (i.e., 25th percentile) $\simeq r'_{0.25|P|}$ of the sorted list.

Result

In order to compute the performance of the composition algorithms with free parameters, we estimated the optimal parameter values using a leave-one-out cross-validation procedure. The cosine similarity and the rank of each compound-word pair was calculated, with the optimal parameters estimated using all the remaining pairs as training data. The parameter m (for PRED) was optimized over integers ranging between 1 and 50 and between 100 and 500 in steps of 50, the parameter k (for PRED and COMP) over integers ranging between 1 and 10, the parameter λ (for DILA) over real numbers ranging between 1.1 and 10.0 in steps of 0.1, and the parameter α (for MULT+WEI) over real numbers ranging between 0.01 and 1.00 in steps of 0.01. The composition algorithms we compared in the experiment are listed in Table 1 for reference. Note that two non-compositional methods (i.e., HEAD and VECT) are considered for purpose of comparison.

Table 2 shows correlation coefficients between human similarity rating and cosine similarity computed in the three semantic spaces. Concerning the superiority between the additive model and the multiplicative model, the result is almost consistent with our hypothesis. In the PPMI-based space, the multiplication algorithm (MULT) performs better than the additive models (i.e., CENT, DILA, PRED, and

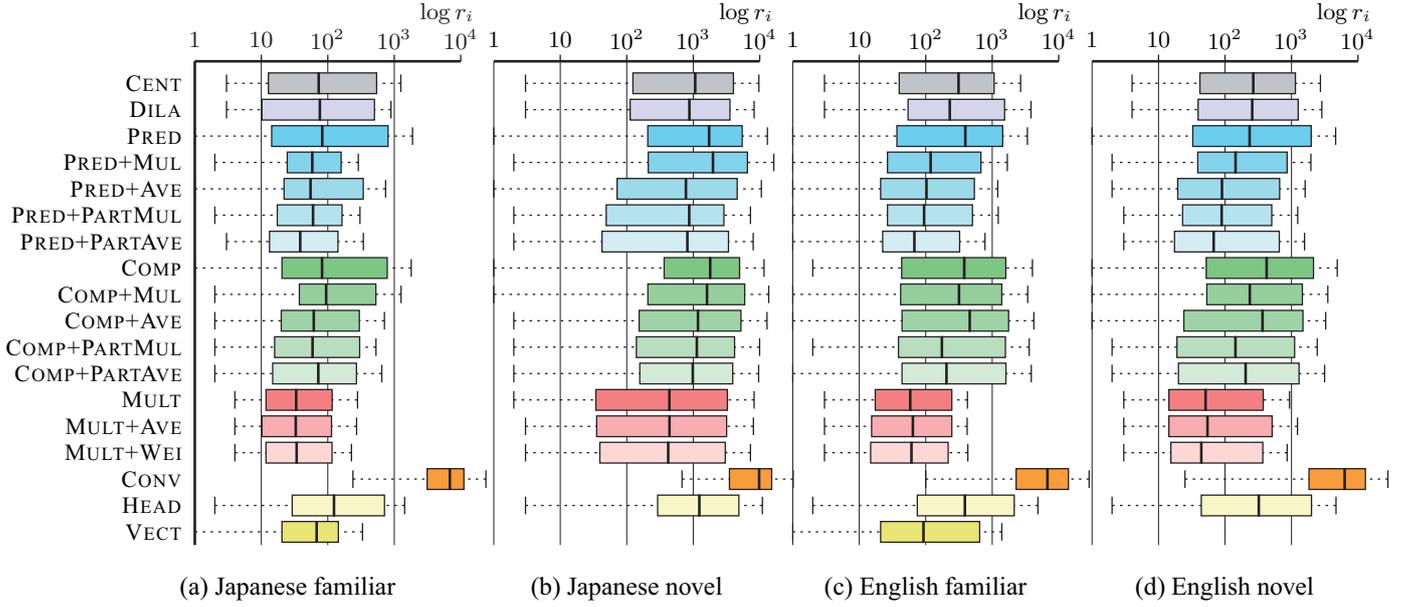


Figure 1: Boxplots of the rank of cosine similarity for compound-word pairs computed in the PPMI-based semantic space

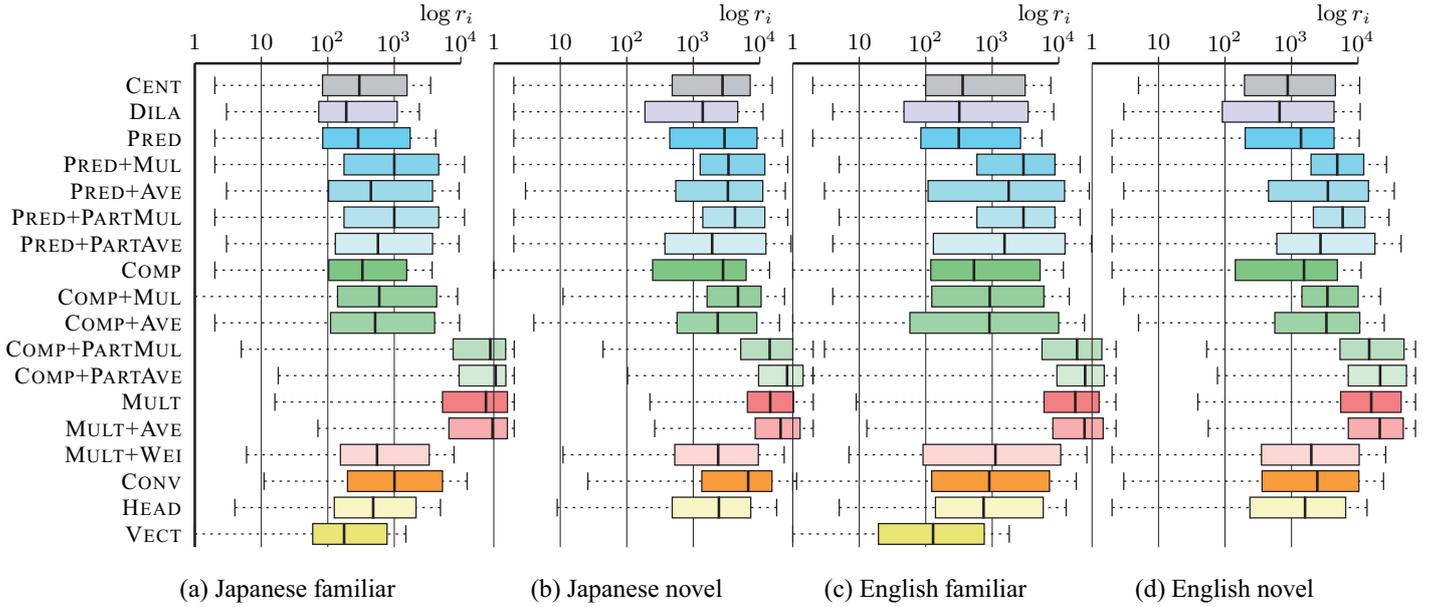


Figure 2: Boxplots of the rank of cosine similarity for compound-word pairs computed in the LSA-based semantic space

COMP), although for familiar compounds it performs worse than the predication algorithm. On the other hand, in the LSA-based semantic space, the multiplication algorithm does not achieve a significant correlation with human judgment, while the additive algorithms are significantly correlated with human judgment. In addition, the smoothed PPMI space (i.e., PPMI+SVD) shows the same tendency, thus suggesting that smoothing by SVD disables the function expected by the component-wise multiplication regardless of the initial matrix and weighting function.

Concerning the proposed algorithms, the extended multiplication algorithms (i.e., MULT+AVE, MULT+WEI) can-

not improve the performance in the PPMI-based space, but the weighted multiplication (MULT+WEI) considerably improves the performance for the semantic space without dimension semantics (i.e., LSA and PPMI+SVD). This surprising finding suggests that moderate weighting enables the multiplicative model to work in these spaces.

The predication and comparison algorithms are improved by some modification methods, especially by multiplying all the vectors (i.e., +MUL), when novel compounds are considered. Furthermore, some modifications also improve the performance for the semantic spaces whose dimensions do not represent distinctive features; the predication algorithm per-

forms best for familiar compounds in the LSA-based space by geometrically averaging all the vectors (i.e., +AVE), and the averaged comparison algorithm (COMP+AVE) performs nearly best for familiar compounds in the smoothed PPMI space. Simple multiplication (+MUL) also improves the performance, but the partial extensions (i.e., +PARTMUL, +PARTAVE) seem not to improve the performance.

Figures 1 and 2 respectively show the results of the similarity ranking of compound-word pairs for the PPMI-based and LSA-based spaces. The result is almost consistent with the result of the correlation analysis. In the PPMI-based space, the multiplication algorithm (MULT) performs better than the additive models, while in the LSA-based spaces it performs much worse than the additive models. The extensions of the multiplication algorithm do not improve the performance in the PPMI-based space, but the weighted multiplication considerably improves the performance in the LSA-based space. The result of the multiplicative modifications (i.e., +MUL, +AVE, +PARTMUL, +PARTAVE) for the predication and comparison algorithms slightly differs between the word ranking test and the correlation analysis. In the word ranking, the multiplicative modifications consistently improve the performance in the PPMI-based space, although the correlation analysis shows that only some modifications do. Most of these improvements observed in the PPMI-based space are not obtained in the LSA-based space. Note that, although not shown in this paper due to lack of space, the results of the smoothed PPMI space do not significantly differ from the results of the LSA-based space.

Discussion

The obtained finding confirms our working hypothesis that the component-wise multiplication works effectively in the semantic spaces whose dimensions represent distinctive features; when reduced by SVD, the dimension of semantic spaces itself loses a semantics, and thus, the component-wise multiplication does not work in those spaces. It follows from this hypothesis that the modification of the additive models by replacing the vector addition with the component-wise multiplication also does not work in those spaces. The experiment demonstrates that in most cases it is true. Although in some cases the multiplicative modification appears effective in the smoothed space, we have no idea whether it is a robust finding or an artifact of a particular experimental setting. Furthermore, the multiplicative modification improves the additive models in the space with dimension semantics, but its degree is not so high. It is interesting for further research to explore in detail the possibility of improving the additive models using the multiplicative model.

A more interesting finding is that the multiplicative model itself can greatly improve by the simple weighting scheme, although the improved performance does not always exceed the performance of the additive models. We are developing a more sophisticated method for extending the component-wise multiplication so that the multiplicative model works in any semantic space.

Acknowledgment

This study was supported by the Grant-in-Aid for Scientific Research B (No.23300098) from the Japan Society for the

Promotion of Science.

References

- Baroni, M., & Zamparelli, R. (2010). Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In *Proceedings of the 2010 conference on empirical methods in natural language processing (emnlp2010)* (pp. 1183–1193).
- Bullinaria, J., & Levy, J. (2007). Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39(3), 510–526.
- Jones, M. N., Kintsch, W., & Mewhort, D. J. (2006). High-dimensional semantic space accounts of priming. *Journal of Memory and Language*, 55, 534–552.
- Kintsch, W. (2001). Predication. *Cognitive Science*, 25(2), 173–202.
- Landauer, T., McNamara, D., Dennis, S., & Kintsch, W. (2007). *Handbook of latent semantic analysis*. Mahwah, NJ: Lawrence Erlbaum Associates.
- Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, 104, 211–240.
- McNamara, D. S. (2011). Computational methods to extract meaning from text and advance theories of human cognition. *Topics in Cognitive Science*, 3, 3–17.
- Mitchell, J., & Lapata, M. (2008). Vector-based models of semantic composition. In *Proceedings of acl-08: HLT* (pp. 236–244).
- Mitchell, J., & Lapata, M. (2010). Composition in distributional models of semantics. *Cognitive Science*, 34, 1388–1429.
- Padó, S., & Lapata, M. (2007). Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2), 161–199.
- Quesada, J. (2007). Creating your own LSA spaces. In T. Landauer, D. McNamara, S. Dennis, & W. Kintsch (Eds.), *Handbook of latent semantic analysis* (pp. 71–85). Mahwah, NJ: Lawrence Erlbaum Associates.
- Recchia, G., & Jones, M. N. (2009). More data trumps smarter algorithms: Comparing pointwise mutual information with latent semantic analysis. *Behavior Research Methods*, 41, 647–656.
- Turney, P. D., & Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37, 141–188.
- Utsumi, A. (2011). Computational exploration of metaphor comprehension processes using a semantic space model. *Cognitive Science*, 35(2), 251–296.
- Zanzotto, F. M., Korkontzelos, I., Fallucchi, F., & Manandhar, S. (2010). Estimating linear models for compositional distributional semantics. In *Proceedings of the 23rd international conference on computational linguistics (coling-2010)* (pp. 1263–1271).

Towards Pinpointing the Neural Correlates of ACT-R: a Conjunction of Two Model-Based fMRI Analyses

Jelmer P. Borst (jelmer@cmu.edu) & John R. Anderson (ja@cmu.edu)

Dept. of Psychology, Carnegie Mellon University

Keywords: model-based fMRI; cognitive modeling; ACT-R.

Introduction

The aim of this study is to provide a precise mapping of five ACT-R modules on the brain. While there exists a predefined mapping between ACT-R modules and brain regions (e.g., Anderson et al., 2007), these regions are relatively crude (cubes of $\sim 12 \times 12 \times 12$ mm) and serve in principle only as indicators of module activity. Previously, we have shown that an analysis method called *model-based fMRI* can provide more detailed whole-brain mappings (Borst, Taatgen, & Van Rijn, 2011). In the current study, we applied this method to a second dataset and combined the results of both datasets to create an overall mapping.

Method

In a typical fMRI analysis, the condition structure of the experiment is regressed against the fMRI measurements. This results in brain areas that are active in response to the experimental conditions. In model-based fMRI, predictions of a model are used as a regressor instead, showing brain areas that correlate with activity of model components (e.g., Gläscher & O'Doherty, 2010).

Recently, we used this method to locate brain areas that correspond to the ACT-R modules in a relatively complex

multitask experiment (Borst et al., 2011). We now applied the same method to analyze a dataset that was published by Anderson et al. (2007), who used a more traditional laboratory experiment to show differential brain activity of eight ACT-R modules.

Naturally, the results of both datasets are partly dependent on idiosyncrasies of the respective models and experiments. We therefore subsequently combined the results of the two datasets with a conservative conjunction analysis (Nichols et al., 2005) to create a more stable mapping.

Results

Figure 1 illustrates the process for the manual module. The left panel shows the results of the model-based analysis of the Anderson et al. (2007) dataset: a region specific to the motor cortex, overlapping with ACT-R's predefined region. The middle panel shows the results of the Borst et al. (2011) dataset: the manual activity of the model correlated most significantly with a region in the visual cortex. In addition, an area in the motor cortex also correlated with the model predictions. The right panel shows the conjunction of both datasets: an area that is constrained to the motor cortex.

Figure 2 shows the results of the conjunction analyses of the other four modules. The aural module correlated with a region in the auditory cortex. The visual module correlated

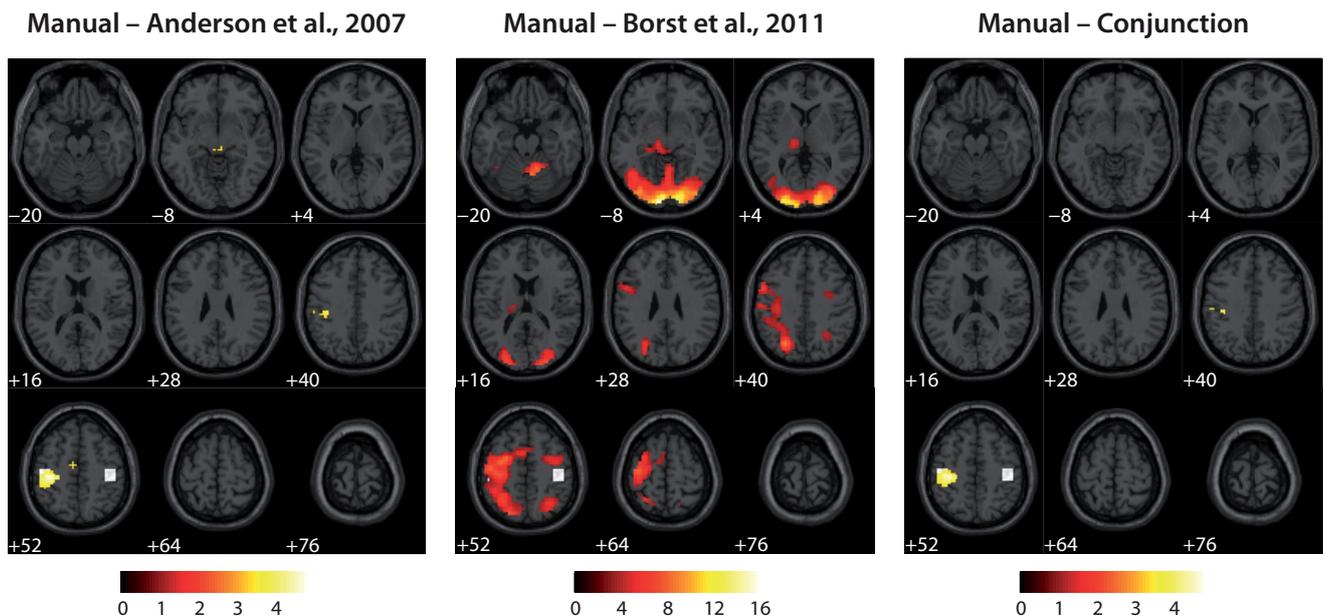


Figure 1. Results for the manual module of the Anderson et al. dataset (left), the Borst et al. dataset (center), and the conjunction (right). All results thresholded at $p < .001$. White squares indicate ACT-R's predefined manual module.

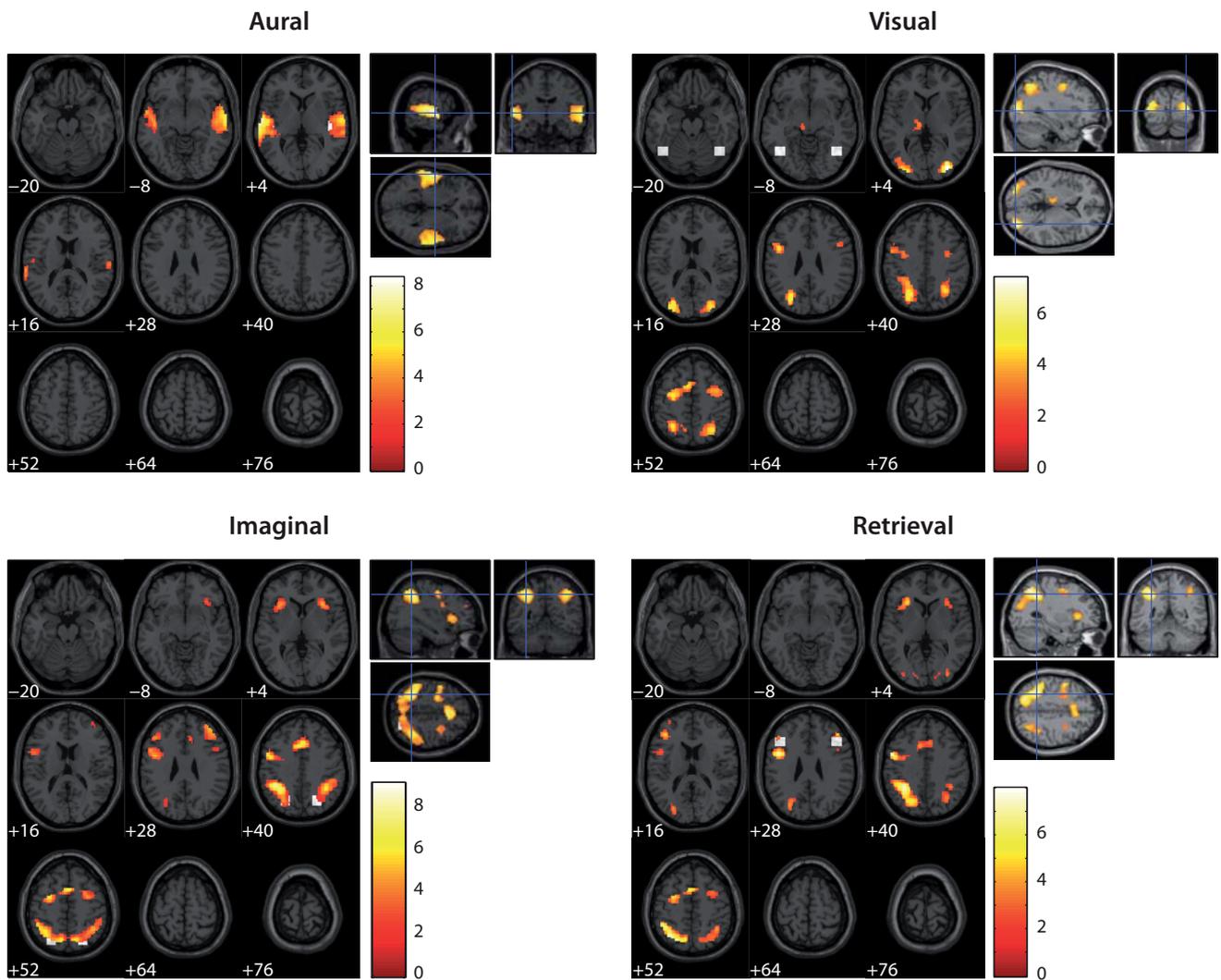


Figure 2. Conjunction results for the aural, visual, retrieval, and imaginal modules, thresholded at $p < .001$. Crosshairs in the ‘3D’ images indicate the most significant voxel. White squares indicate the respective predefined ACT-R modules.

most with an area in the visual cortex, but also with other areas. As expected, the retrieval module correlated with a prefrontal area, but most strongly with the parietal region that is normally attributed to the imaginal module. Finally, the imaginal module correlated most with the expected area in the parietal cortex, but also with other areas.

Discussion

Figure 1 highlights the strength of the methodology: while the Borst et al. analysis (2011) resulted in the visual cortex, the combination of the two datasets led to the correct motor region. On the other hand, for the retrieval module we did not find the expected prefrontal region as the best matching area, and for most modules we found activity throughout the brain. While these results might turn out to be the right representations of the modules, it seems to suggest that we should add more datasets to the analysis to create more constrained mappings. In general, combining model-based

fMRI results of multiple experiments seems to be a very promising method to locate the neural correlates of ACT-R.

References

- Anderson, J. R., Qin, Y., Jung, K. J., & Carter, C. S. (2007). Information-processing modules and their relative modality specificity. *Cogn. Psychol.*, *54*(3), 185-217.
- Borst, J. P., Taatgen, N. A., & Van Rijn, H. (2011). Using a symbolic process model as input for model-based fMRI analysis: Locating the neural correlates of problem state replacements. *NeuroImage*, *58*(1), 137–147.
- Gläscher, J. P., & O’Doherty, J. P. (2010). Model-based approaches to neuroimaging: Combining reinforcement learning theory with fMRI data. *WIREs Cogn Sci*, *1*(4), 501-510.
- Nichols, T., Brett, M., Andersson, J., Wager, T., & Poline, J.-B. (2005). Valid conjunction inference with the minimum statistic. *NeuroImage*, *25*(3), 653-660.

Qualitative and Quantitative Individual Differences in Semantic Categorization

Steven Verheyen (steven.verheyen@ppw.kuleuven.be)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 - box 3711, B-3000 Leuven, Belgium

Gert Storms (gert.storms@ppw.kuleuven.be)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 - box 3711, B-3000 Leuven, Belgium

Keywords: mixture models; IRT models; threshold models.

Introduction

When asked to indicate which items from a set of candidates belong to a particular category, inter-individual differences appear: Individuals disagree on the items that should be considered category members (e.g., Black, 1937; Hampton, Dubois, & Yeh, 2006; McCloskey & Glucksberg, 1978). Individuals might disagree about whether *hiking* and/or *darts* are *sports*, for instance. We will argue that inter-individual differences in semantic categorization come in two kinds. (i) Qualitative differences reflect a different organization of the candidate items with respect to the target category. (ii) Quantitative differences reflect a different propensity to endorse items as category members.

Qualitative differences represent different views on what are considered representative category members. Individuals who consider *hiking* a better example of *sports* than *darts* presumably find that the former meets the requirements of category membership better than the latter does. They, for instance, recognize that *hiking* is physically more demanding than *darts* is. Individuals who consider *darts* to be the better example must then employ different requirements for category membership. When judging category membership they place more emphasis on elements such as rules or competition, for instance. These requirements are better met by *darts* than by *hiking*.

Among individuals who agree on the organization of items with respect to the target category, categorization differences of a quantitative nature can arise. These are differences that pertain to the propensity to endorse items as category members. The item organization reflects the varying extents to which the items fulfill the requirements for category membership (e.g., *hiking* is physically more demanding than *darts* is). Certain individuals might want to see more evidence of these requirements than others. They might only deem *hiking* physical enough to be considered a *sport*, while others find both *darts* and *hiking* demanding enough.

Model

Our goal is to elucidate these two kinds of inter-individual categorization differences by means of the mixture item response theory model (Mislevy & Verhelst, 1990; Rost, 1990) that is expanded in Equation (1). Models like these are traditionally employed to assess individuals' aptitudes and dispo-

sitions in response to a number of test items. However, they have also been shown to be flexible tools to analyze semantic categorization data (e.g., Verguts, De Boeck, & Storms, 1998; Verheyen, Hampton, & Storms, 2010).

Here the model will be used to partition a large sample of categorizers in a number of groups that are maximally different in terms of their organization of the items with respect to the target category. These item organizations take the shape of scales along which all candidate items are positioned according to their likelihood of being endorsed. Different organizations capture qualitative categorization differences between participants from different groups: They reveal how a particular item might be a likely category member in one group, but an unlikely category member in another group. Participants that end up together in a group are understood to adopt the same item organization. These categorizers do not differ qualitatively, but can display varying degrees of propensity to endorse items as category members. These are inter-individual categorization differences of a quantitative nature. In the formal framework they take the shape of criteria that are imposed on the scales that organize the candidate items: They reveal how some individuals in a group might use very liberal criteria, while others employ very stringent criteria.

Binary categorization decisions Y_{ci} constitute the input for the model. Here the categorization data are comprised of member/non-member decisions Y by 250 categorizers c towards 24 items i in each of 8 natural language categories (*fish, fruits, furniture, insects, sciences, sports, tools, vegetables*).

Every one of these categorization decisions is considered the outcome of a Bernoulli trial with the probability of a *member* response:

$$\Pr(Y_{ci} = 1) = \frac{e^{\alpha_g(\beta_{g,i} - \theta_{gc})}}{1 + e^{\alpha_g(\beta_{g,i} - \theta_{gc})}} \quad (1)$$

In Equation (1) the betas capture the organization of the items with respect to the target category. g groups of categorizers are extracted, with separate item organizations that are maximally different. For each group the organization takes the shape of a scale along which all candidate items are positioned. $\beta_{g,i}$ indicates the position of item i along the scale for group g . Higher values for $\beta_{g,i}$ indicate likelier category members.

The thetas in Equation (1) capture the degree of liberalism/conservatism categorizers display. A separate indication of the propensity to endorse items as category members is extracted for each categorizer c . It takes the shape of a criterion that is positioned along the same scale that organizes the items for the group the participant belongs to. θ_{gc} indicates the position of the criterion for categorizer c along the scale for group g . Higher values for θ_{gc} indicate more conservative categorizers.

Unlike the betas and thetas, the alphas in Equation (1) can only take on positive values. A separate α_g for each group determines the shape of the response function that relates the unbounded difference $\beta_{g,i} - \theta_{gc}$ to the probability of a *member* response (bounded between 0 and 1). Indeed, the relative position of item and criterion along a scale determines the probability of a *member* response. If $\beta_{g,i}$ equals θ_{gc} the numerator of Equation (1) takes on the value of 1, while the denominator takes on the value of 2. The resulting probability is .50, indicating that the categorization decision can go either way. The odds change when item and criterion have a different position along the scale. If the item surpasses the criterion, the odds are that the categorizer will endorse it. The greater the distance between item and criterion, the greater the odds of a *member* decision. If the item does not surpass the criterion, the odds are that c will not endorse i . Under this circumstance, the odds of a *non-member* decision increase with the distance between item and criterion.

The one-group variant of the model in Equation (1) has been applied to semantic categorization by Verheyen et al. (2010). That particular model only allows for quantitative inter-individual categorization differences. Participants can differ in terms of the categorization criterion they employ, but not in terms of the scale along which the criteria are placed. They all adopt the same category organization. The model in Equation (1) is more general. It allows for qualitative differences in addition to quantitative ones. It relaxes the assumption that all participants adhere to the same category organization. Instead, it assumes that the participants divide in groups with a different item organization each. (One set of beta estimates is extracted for each group.) Within each group, individuals are still thought to differ in terms of the employed categorization criterion. (A theta estimate is extracted for every categorizer.) The model in Equation (1), then, is a mixture of differently parameterized quantitative differences-only models of the kind employed by Verheyen et al. (2010).

Findings

The analysis of the categorization data with the mixture model in Equation (1) yields evidence for both qualitative and quantitative inter-individual differences. For the categories of *fish*, *insects*, *sciences*, *sports*, and *tools* the sample of categorizers divides in distinct groups, who regard different items likely category members (i.e., qualitative differences). Within each of these groups categorizers differ in their propensity to provide membership responses (i.e., quan-

titative differences). The existence of multiple item organizations for a single category suggests that it might be improper to assume a default category representation that is the same for all language users. Rather, it would appear that there exist a number of these default representations, which emphasize different sets of category features. Indeed, a clear pattern emerged when we (i) determined to what extent features that participants consider important for category membership are true of the different candidate items, (ii) obtained a small number of principal components that convey the information that is contained in these feature applicability judgments, and (iii) regressed the item organizations of different groups upon these principal components. For each of the categories with multiple item organizations, there was at least one component that had a similar effect on every item organization. Common components indicate agreement among groups on what it means to be a category member. This is required for members of different groups to successfully communicate with one another using the studied natural language terms. The item organizations could also be distinguished on the basis of other components that were of importance to single subgroups only. These distinct components indicate disagreement among groups on what it means to be a category member but do not appear to hamper communication between members of different groups.

Acknowledgments

Steven Verheyen is a postdoctoral fellow at the Research Foundation-Flanders.

References

- Black, M. (1937). Vagueness: An exercise in logical analysis. *Philosophy of Science*, 4, 427-455.
- Hampton, J. A., Dubois, D., & Yeh, W. (2006). Effects of classification context on categorization in natural categories. *Memory & Cognition*, 34, 1431-1443.
- McCloskey, M. E., & Glucksberg, S. (1978). Natural categories: Well defined or fuzzy sets? *Memory & Cognition*, 6, 462-472.
- Mislevy, R. J., & Verhelst, N. (1990). Modeling item responses when different subjects employ different solution strategies. *Psychometrika*, 55, 195-215.
- Rost, J. (1990). Rasch models in latent classes: An integration of two approaches to item analysis. *Applied Psychological Measurement*, 14, 271-282.
- Verguts, T., De Boeck, P., & Storms, G. (1998). Analyzing experimental data using the Rasch model. *Behavior Research Methods*, 30, 501-505.
- Verheyen, S., Hampton, J. A., & Storms, G. (2010). A probabilistic threshold model: Analyzing semantic categorization data with the Rasch model. *Acta Psychologica*, 135, 216-225.

Modeling Efficiency-guided Modality Choice in Voice and Graphical User Interfaces

Stefan Schaffer (sschaffer@zmms.tu-berlin.de)
Berlin Institute of Technology
Research Training Group prometei, Franklinstr. 28-29
10827 Berlin, Germany

David Reitter (reitter@cmu.edu)
Carnegie Mellon University
Department of Psychology, 5000 Forbes Avenue
Pittsburgh, PA 15213 USA

Keywords: Multimodal interaction; input modality choice; strategy selection.

Introduction

In multimodal human computer interaction users can often select between specific input modalities. Modality choice is influenced by various factors including user attributes, system attributes, the task and the environment (e.g. Lemmelä et al., 2008). Here, we describe on-going research into cognitive models of input modality selection.

The efficiency to solve a task with a multimodal user interface can vary widely due to modality-specific shortcuts. For instance, comparing touch-screen and speech input, items in lists such as names in a directory can be more efficiently found via speech. The number of list items in a GUI is limited due to screen size and legibility. Using a touch-screen, users have to browse the list for the searched item. With speech, each item can be directly accessed, as the limitations of the GUI do not necessarily affect the voice interface. Thus, subjects find a list item in fewer steps by asking for it verbally. The benefit of speech, B_S , is defined as the difference in interaction steps between touch-screen (IS_T) and speech inputs (IS_S): $B_S = IS_T - IS_S$.

Our aim is to develop models of modality selection to support existing tools for model-based usability evaluation such as MeMo (Möller et al., 2006) or CogTool (John et al., 2004). In a classical usability experiment, a participant is instructed to solve a task with different user interface variants. Taatgen et al. (2006) presented a model where unimodal task knowledge was coded into instructional chunks of the declarative memory of ACT-R (Anderson et al., 2004). We extended Taatgen's concept for multimodal interaction and investigate to which extent our model is able to reproduce the modality selection behavior of real test participants.

Experiment

The Restaurant Booking System (RBS)

A smart phone-based RBS with touch and speech as input modalities was tested (for details, see Schaffer, 2011a). Automatic speech recognition (ASR) was simulated via a Wizard-of-Oz design: an unseen human operator changed the system state. This way, issues related to ASR errors could be avoided.

In the RBS database, requests consisting of a name of a city, a culinary category, a desired time and the number of people are made. All user entries are entered via different lists. Each list contains 6 layers each with 4 items. The

transition between layers is performed with touch or speech input. An item is selected by touch or by saying the written text label. However, all list items can also be accessed directly by using speech input. The items are ordered alphabetically or numerically. The benefit of speech input calculates to 0 steps ($B_S = 1 - 1$) for items located at the first layer of a list and increases to 5 steps ($B_S = 6 - 1$) at the last layer of a list.

Task

The participants' task was to perform database requests with the RBS. The benefit of the speech modality was systematically varied between 0 and 5 interaction steps.

Participants

Sixteen German-speaking participants (8 female, 8 male) between the age of 22 and 31 ($M=26$, $SD=2.95$) took part in the study. A single experiment took approximately one hour. Participants received a remuneration of €10.

Procedure

The system was explained and the usage of touch and speech demonstrated. Then, participants performed three training trials: touch usage only, speech usage only and multimodal with mixed modality usage. In the target phase, 12 trials with mixed, participant-chosen modality usage followed. The tasks were presented in written form (e.g., "Please find a Chinese restaurant in Berlin at 8 pm for 12 people").

Cognitive Model

Instructional steps are represented in declarative memory as chunks containing pre-condition, post-condition, action and modality. Pre- and post-conditions are used to chain the instructions. A key aspect of the model is that for each modality, instructional chunks with the same precondition occur in declarative memory. An earlier study revealed that for the RBS, speech is perceived to be more demanding than touch input (Schaffer 2011b). Therefore we use an action slot within each instruction to describe the interaction more precisely. One GUI interaction consists of two instructions distinguished by the statement in the action slot (search and

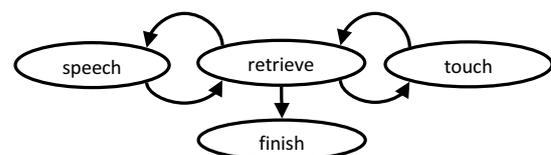


Figure 1: Procedural knowledge of the model.

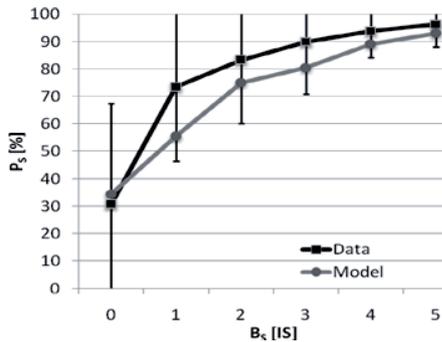


Figure 2: Percentage of speech usage P_S for each level of benefit B_S for human and model data.

press). Speech input consists of three instructions (action slots: search, think and speak).

The general operation of the model is summarized in Figure 1. Instructions are being retrieved from declarative memory. Chunks with the same precondition (but differing modality) are chosen randomly. Retrievals are processed by modality specific production rules. By way of the production compilation mechanism, new production rules with integrated chunks are learned. After each finalization of the task a reward is propagated to the involved productions. Thus, the model adapts to modality success via a reinforcement-learning mechanism.

Results

Figure 2 shows the percentage of speech usage P_S in the human data (black) and the model data (grey). An analysis of variance with repeated measures showed a highly significant effect of B_S on P_S in human data ($F(2.27,33.97)=27.503$; $p_{1-tailed}<.001$; $\text{part.}\eta^2=.647$).

Modality usage of the model is comparable to human behaviour. The model performs fairly well at $B_S=0$. For $B_S=1, 2$ and 3 the model fit worsens, whereas for $B_S=3$ and 4 model performance improves again.

Each participant (16) of the experiment executed eight subtasks for each level of B_S . Thus the model data was calculated from the average of 128 particular model iterations. Each iteration included 150 runs. Figure 3 shows the learning behavior for each level of B_S (colored lines).

Conclusion

Taken in context with our aim to design tools for model-based usability evaluation, the model provides a useful basis for a modality selection mechanism. Future work will extend the model to enable interaction with system prototypes and produce actual speech output. As is seen sometimes in reinforcement learning, adaptation seems slower than what is seen empirically. Once a better-fitting model is defined, further evaluation may demonstrate the learning behavior over repeated presentations and time, giving essential cues to the nature of the learning effect as a form of routinization or declarative memorization. One

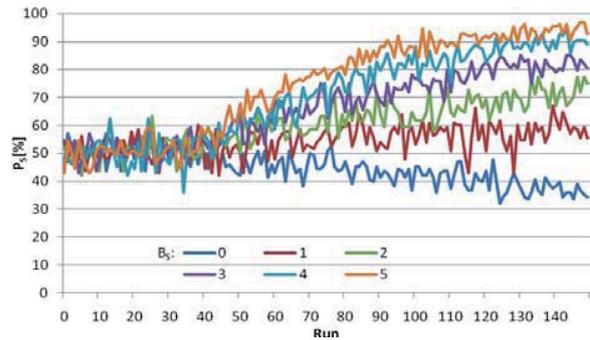


Figure 3: Development of speech usage P_S during 150 model runs for different levels of speech benefit B_S .

common effect of routinization is that early choices and experiences determine fixed, long-term strategy choices as routinized knowledge is less adaptive (an effect of primacy: first impressions matter). Showing such effects would critically examine the use of adaptive speech recognition technology in end-user applications, specifically if these systems start out with high error rates.

Acknowledgements

The research is funded by the German Research Foundation (DFG - 1013 'Prospective Design of Human-Technology Interaction') and supported by Deutsche Telekom Laboratories. We also thank Michael Minge for his help during the experiment.

References

- Anderson, J.R., Bothell, D., Byrne, M., Douglass, D., Lebiere, C. & Qin, Y. (2004). An integrated theory of mind. *Psychological Review*, 111 (4), 1036-1060.
- John, B. E., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). Predictive human performance modeling made easy. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (pp. 455-462). New York, NY, USA: ACM.
- Lemmelä, S., Vetek, A., Mäkelä, K., & Trendalov, D. (2008). Designing and evaluating multimodal interaction for mobile contexts. In *Proceedings of the 10th Int'l Conf. on Multimodal Interfaces* (pp. 265-272). New York, NY, USA: ACM.
- Möller, S., Englert, R., Engelbrecht, K., Hafner, V., Jameson, A., Oulasvirta, A. (2006). Memo: Towards automatic usability evaluation of spoken dialogue services by user error simulations. In *Proc. 9th International Conference on Spoken Language Processing* (p. 1786-1789). Pittsburgh, PA, USA.
- Schaffer, S., Jöckel, B., Wechsung, I., Schleicher, R. & Möller, S. (2011a). Modality Selection and Perceived Mental Effort in a Mobile Application. In *Proc. 12th Annual. Conf. International Speech Communication Assoc.* (p. 2253-2256). Florence, Italy.
- Schaffer, S., Schleicher, R., & Möller, S. (2011b). Measuring cognitive load for different input modalities. In *9. Berliner Werkstatt Mensch-Maschine-Systeme* (p. 287-292). Berlin, Germany: VDI Verlag.
- Taatgen, N. A., Huss, D., & Anderson, J. R. (2006). How cognitive models can inform the design of instructions. In *Proc. 7th Int'l Conf. on Cognitive Modeling* (p. 304-309). Trieste, Italy: Edizioni Goliardiche.

A First Approach to Generate Hybrid Animations for Maintenance Tasks in IPS²

Ulrike Schmuntzsch, Ulaş Yılmaz, Ralf Reichmuth & Matthias Rötting
{usc, uyi, rre, mro}@mms.tu-berlin.de

Department of Psychology and Ergonomics, Chair of Human-Machine Systems
Franklinstr. 28/29, 10587 Berlin

Keywords: human modeling, MOCAP systems, data glove, animated virtual character, instruction video, Make Human, Blender, IPS²

Motivation and Project Objectives

Due to reduced prices and system complexity, Motion Capture (MOCAP) technologies, which traditionally are applied in ergonomics research and film/game industry, nowadays find more and more use in other areas (Bergler 2007, Brodie et al., 2008, XSens 2012). Another new technology, which finds application in an extended set of areas, like online markets or computer games, is the provision of instructions by animated virtual characters in order to successfully support a multitude of different users. Yet, when considering maintenance tasks, there are only few examples (Ziegeler & Zühlke, 2005). A well-known one is the animated pedagogical agent “Steve”, who teaches users how to use or overhaul complex machines (Rickel & Johnson, 1999). According to Rickel (2001) many advantages are united in animated pedagogical agents such as attention guidance with gaze and gestures. The major benefit is a deeper and faster learning process especially of procedural knowledge. This results from synchronizing oral explanation with visualized action sequences and with that a much more facilitated transfer of practical knowledge (Mischas & Berry, 2000).

Regarding the context of *Industrial Product-Service Systems* (IPS²), which are characterized by a variety of different users, contexts of usage and highly specialized products and services an urgent need of descriptive and comprehensible user support is clearly noticeable (Uhlmann et al., 2008, Schmuntzsch & Rötting, 2011). Thus, our IPS²-related project (SFB/ TR29 B4) focuses, inter alia, on the combination of MOCAP technology with the modeling of an animated pedagogical agent, who demonstrates how to perform a maintenance task in an instruction video. In this article we will give a brief overview of the design process and address the advantages and challenges which came up.

Design Process

The overall design process of hybrid animation consists of three consecutive steps explained in the following sections.

Capturing Hand Movements with MOCAP

The first step is to use a data glove to capture hand movements of a human operator replacing a spindle on a micro milling machine as shown in Figure 1.

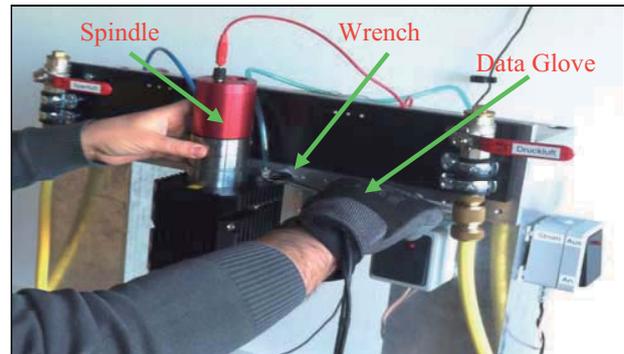


Figure 1: Replacing the spindle of a micro milling machine using the data glove.

There are four typical data glove systems on the market: optical, mechanical, inertial and bend. We chose a system that uses bend sensors so that we could safely acquire the hand movements without the effects of occlusion and the magnetic interference of the metallic parts. We use an *X-IST Wireless DataGlove* that has bend sensors on fingers (X-IST 2009): The sensors are located on each finger, 2 being on the thumb and 3 on each of the rest. Each sensor delivers a raw value in range [0, 1024] that corresponds to the relative bend of the finger bone at that instance. Since these bend values vary depending on the anatomy of the operator’s hand, finding a general mapping of them on to a virtual hand model is a challenging task. Thus, we have implemented an interactive interface through which the operator calibrates the data glove, in order to get a personalized bend value to virtual hand mapping schema. Note that, with *X-IST DataGlove* hand rotation is poorly captured through inertial sensors and finger spread is not to be acquired at all.

Creating the Human Model

The second step of the design process is to build a human model of the animated pedagogical agent by using the software tool *MakeHuman 1.0 alpha6.0*, which is followed by the export of the human model as “Blender Exchange” in *mhx* file format. Besides the human model, the micro milling machine with its spindle and different tools, such as a torque handle, allen key and jaw spanner, were created with the program *Autodesk Inventor 2012* and also exported to a *Blender* suitable Format. Hereafter, all exported models were modified through several small adaptations in order to create a natural and familiar impression on the user (See Figure 2).



Figure 2: Human model as virtual character in *Blender*.

After animating the visual sequences of events of the maintenance task, -in this case the spindle change-, the associated and already recorded oral explanations are integrated in the instruction video (Schmuntzsch, Reichmuth, Sturm & Rötting, to appear).

Creating the Hybrid Animation of MOCAP and the Human Model as Instruction Video

In a third step, the combination process is carried out. i.e. the captured hand movements are mapped to the human model in *Blender*. The hybrid animation obtained through this process combines the easy-to-model and relatively coarse body movements with through data glove acquired and fine hand movements, which brings a considerable amount of realism to the created instruction video. There are two possible ways to merge the hand modeled animation and the animation acquired using the data glove. In the first approach, typical gestures of the hand are acquired and they are inserted as keyframes in the hand modeled animation. This approach is quite appropriate when the hand movements have repeating gesture patterns which do not vary much over time. In the second approach, dynamic action sequences are acquired and mapped on the overall animation by scaling over time. This approach enables modeling complex hand movements; however, it increases the number of introduced keyframes. In our study we use both approaches depending on the action under consideration: Grasping a tool is acquired as an action sequence whereas the hand state before and after the grasp action is acquired as a static hand gesture.

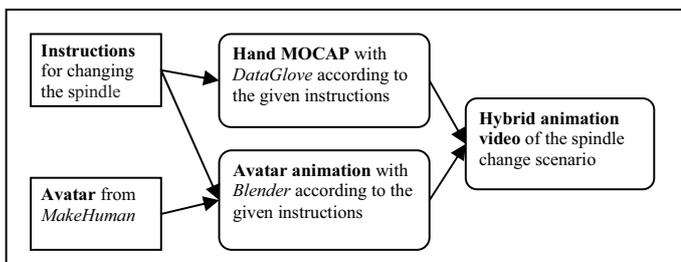


Figure 3: Overall system architecture.

Conclusions and Future Work

This article describes the first steps in generating a hybrid animation for maintenance tasks in IPS². Starting off by capturing hand movements with a data glove, creating a human model followed. Then, we create a hybrid animation as an instruction video by combining the MOCAP and the human model. Overall system flow is illustrated in Figure 3. Even though these technologies are used in the film industry for quite a long time, IPS² is a rather new field of application. Growing product complexity and increasing heterogeneity of users and contexts highlight the importance of understandable and illustrative user support such as video instructions. Since maintenance tasks in IPS² require a great amount of dexterity, MOCAP systems allow exact visualization of single finger movements. Currently, the captured hand movements are combined with a full-body animated human model. However, we are striving for an integration of a data glove and a full-body motion suit to create instruction videos for maintenance tasks.

References

- Bergler, C. (2007). Motion Capture Technology for Entertainment. *IEEE Signal Processing Magazine* 24(6) (pp. 160-158).
- Brodie M., Walmsley A. & Page W. (2008). Fusion motion capture: a prototype system using inertial measurement units and GPS for the biomechanical analysis of ski racing. *Sports Technologies* 1(1) (pp. 17-28).
- Mischas, I. C. & Berry, D. (2000). Learning a procedural task: effectiveness of multimedia presentations. *Applied Cognitive Psychology*, 14, 555–575.
- Uhlmann, E., Meier, H, Stelzer, C., Geisert, C., Bochnig, H, & Sadek, K. (2008). Design of PSS Based on Customer Requirements, First International Seminar on PSS, Bochum, Shaker; Auflage: 1 (Juni 2008), ISBN-10: 3832272275.
- Rickel, J.& Johnson, W.L. (1999). Animated agents for procedural training in virtual reality: Perception, cognition, and motor control. *Applied Artificial Intelligence* 13 (pp. 343-382).
- Rickel, J. (2001). Intelligent Virtual Agents for Education and Training: Opportunities and Challenges. In A. de Antonio, R. Aylett, and D. Ballin (Eds.), *IVA 2001, LNAI 2190*. (pp. 15-22). Springer-Verlag Berlin Heidelberg.
- Schmuntzsch, U., Reichmuth, R., Sturm, C. & Rötting, M. *Virtual Agent Assistance for Maintenance Tasks in IPS²*. [To appear in the 4th Int. Conf. on Applied Human Factors and Ergonomics, July 2012].
- Schmuntzsch, U. & Rötting, M. (2011). *Multimodal User Interfaces in IPS²*. In V.G. Duffy (Ed.), *Digital Human Modeling, Lecture Notes in Computer Science*, Vol. 6777 (pp. 347-356). Berlin: Springer.
- X-IST (2009). X-IST DataGlove Wireless. <http://www.vrealities.com/x-ist.html> (February 21, 2012).
- XSens (2012). XSens MVN Customer Examples. www.xsens.com/xsens-mvn-examples (February 21, 2012).
- Ziegeler, D. & Zuehlke, D. (2005). Emotional user interfaces and humanoid avatars in industrial environments. *Proceedings of the 16th IFAC World Congress*, 2005.

Abstraction in Natural Language Categories

Loes Stukken (loes.stukken@ppw.kuleuven.be)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 - box 3711, B-3000 Leuven, Belgium.

Wouter Voorspoels (wouter.voorspoels@ppw.kuleuven.be)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 - box 3711, B-3000 Leuven, Belgium.

Gert Storms (gert.storms@ppw.kuleuven.be)

Laboratory of Experimental Psychology, University of Leuven
Tiensestraat 102 - box 3711, B-3000 Leuven, Belgium.

Wolf Vanpaemel (wolf.vanpaemel@ppw.kuleuven.be)

Quantitative Psychology and Individual Differences, University of Leuven
Tiensestraat 102 - box 3713, B-3000 Leuven, Belgium

Keywords: formal models; categorization; concepts; typicality.

Introduction

A long lasting debate in research on the fundamental human capacity to categorize, regards the degree of abstraction in the mental category representations of a cognitive agent. In its extremes, this debate translates into the decades-old discussion between advocates of the exemplar view (Nosofsky, 1984) and the prototype view (Minda & Smith, 2001). Recently, a number of models have been developed that go beyond these extremes (Griffiths, Canini, Sanborn, & Navarro 2007; Love Medin & Gureckis, 2004; Rosseel, 2002; Vanpaemel & Storms, 2008). One of them is the Varying Abstraction Model (VAM) proposed by Vanpaemel and Storms (2008) in which a whole range of category representations (including the prototype and exemplar representation) can be tested and compared with one another. In this way, the VAM can be used to quantify the degree of abstraction in artificial categories.

Surprisingly, these and other formal models that have been extensively used to investigate the category representations of artificial categories are rarely used to study the category representations of natural language categories such as *fruits* and *birds*. This is odd, since ultimately the goal of investigating artificial categories is to better understand how people learn and use everyday concepts, that is, natural language categories. Given that natural language categories can be expected to be different from artificial categories in a number of respects (Malt & Smith, 1984), the results from artificial categorization experiments may not easily generalize to natural language categories

In the present study we test whether we can find evidence for partial abstraction in natural language categories. We adapt the VAM to make it applicable to the domain of natural language categories and test it in two categories: *fruits and birds*.

The Varying Abstraction Model

The VAM starts from the assumption that the prototype and exemplar representation are two extremes on a continuum ranging from maximal abstraction (prototype) to minimal abstraction (exemplar) and furthermore states that besides these two extremes also intermediate representations on this continuum should be considered as valuable category representations. These intermediate category representations correspond to representations in which some exemplars are merged to form a set of prototypes and where other exemplars can be represented individually.

The category representations of the VAM are formed by subprototypes. To define subprototypes, VAM uses a multidimensional space to represent the exemplars of a category. Subprototypes are formed by dividing the points, that make up a category in the MDS space, in clusters and by averaging the coordinates of the points that were clustered together.

The more subprototypes that make up a representation the less abstract the representation is. The least abstract representation of the VAM, is the exemplar representation for which no exemplars are merged together. If all the exemplars are merged in one cluster the obtained representation is the most abstract representation of the VAM namely, the prototype representation and a representation with two subprototypes is, for example, slightly less abstract than a prototype representation but still more abstract than an exemplar representation.

Given a category representation, the VAM uses the processes of the well-known Generalized Context Model (GCM) of Nosofsky (1984) to determine the category decisions a subject makes for a particular stimulus. The VAM derives, in the same way as the GCM, similarities from the MDS space and uses, like the GCM, the Luce choice rule to derive category decisions from these similarities. The only difference is that the VAM contains not only the exemplar representation but also the prototype

and all the possible intermediate representations lying between the prototype and exemplar representation.

Vanpaemel and Storms (2008,2010) fitted the VAM to category decisions made for artificial categories and showed that an intermediate representation provided a better fit to the data in some of the artificial categories they studied, suggesting that these intermediate category representations are valuable category representations for artificial categories.

Varying abstraction in natural language categories

When applying the VAM to the domain of natural language concepts, two considerations are in place. First, whereas in category learning experiments with artificial stimuli the dependent variable typically is a categorization decision, this variable seems rather awkward for semantic concept research since people are generally in good agreement of the exemplars that belong to a particular category and those that do not. Category decisions are, therefore, not the primary variable studied in natural language categories. Researchers investigating natural language categories usually use typicality as a dependent variable in their studies. Typicality is a measure of how good an exemplar is an example of a category. Typicality has been used extensively in the studies about the category representations of natural language categories and is known to predict performance in a variety of cognitive tasks (for a review see Hampton, 1993).

In order to obtain typicality predictions for each exemplar from the representations of the VAM model we calculate the similarity of the exemplars to the category, which in case of the VAM corresponds to summing the similarity of the exemplar to all the subprototypes that make up the category.

Second, everyday concepts have an extension that greatly outnumbers the largest artificial categories generally studied. This greatly increases the complexity of studying varying abstraction in these categories. There is no a priori restriction in the VAM in the way that exemplars of a category are merged into subprototypes. This means that in a category with an extensive number of exemplars the number of possible category representations quickly becomes untenable. A category with 30 exemplars for example yields $8.4675 \cdot 10^{23}$ different category representations. One way to solve this issue is to assume that some category representations are more plausible than others. It is for example much more likely that similar members of a category will be clustered together in a category representation while dissimilar members will be kept separate. This idea is elegantly captured by applying k-means clustering, in which similar members are assigned to the same cluster and dissimilar members are kept separate from each other. By using k-means clustering we were able to select a single category representation for every number of subprototypes.

Data

In our study we investigated the natural language categories *fruits* and *birds* with respectively 44 and 41 exemplars. To

construct an MDS space for each category we gathered pairwise similarity ratings for each category by asking four subjects to rate the similarity between each pair of exemplars on a scale from 1 (not at all similar) to 9 (very similar). Typicality ratings for each exemplar were obtained by asking subjects to rate the typicality of the exemplars on a scale from 1 (not at all typical for the category) to 20 (very typical for the category).

Results

For each of the categories, we optimized the correlation between observed and model-based typicality scores for each representation at each level of abstraction separately. This results in 44 model fits for the category *fruits* and 41 model fits for the category *birds*.

The results of the model analyses will be discussed in the light of earlier findings.

References

- Griffiths, T. L., Canini, K. R., Sanborn, A. N., & Navarro, D. J. (2007). Unifying rational models of categorization via the hierarchical Dirichlet process. In D. S. McNamara & J. G. Traffon (Eds.), *Proceedings of the 29th Annual Conference of the Cognitive Science Society* (pp. 323-328). MahWah, NJ: Erlbaum.
- Hampton, J. A. (1993). Prototype models of concept representation. In I. Van Mechelen, J. A. Hampton, R. S. Michalski, & P. Theuns (Eds.), *Categories and concepts: Theoretical views and inductive data analysis* (pp. 67-95). London: Academic Press.
- Love, B. C., Medin, D.L., & Gureckis, T. M. (2004). SUSTAIN:A network model of category learning. *Psychological Review*, 111, 309-332.
- Malt, B.C., & Smith, E.E. (1984). Correlated properties in natural categories. *Journal of Verbal Learning and Verbal Behaviour*, 23, 250-269.
- Minda, J.P., & Smith, J.D. (2001). Prototypes in category learning: The effects of category size, category structure, and stimulus complexity. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 27, 775-799.
- Nosofsky, R.M. (1984). Choice, similarity, and the context theory of classification. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 10,104-114.
- Rossee, Y. (2002). Mixture models of categorization. *Journal of Mathematical Psychology*, 46, 178-210.
- Vanpaemel, W. & Storms, G. (2008). In search of abstraction: the varying abstraction model of categorization. *Psychonomic Bulletin and Review*, 15, 732-749.
- Vanpaemel, W. & Storms, G. (2010). Abstraction and model evaluation in category learning. *Behavior Research Methods*, 42 , 421-437.

Semantic Cognition: A Re-examination of the Recurrent Network “Hub” Model

Olivia Guest (O.Guest@bbk.ac.uk) and Richard P. Cooper (R.Cooper@bbk.ac.uk)

Department of Psychological Sciences, Birkbeck, University of London
Malet Street, London WC1E 7HX, UK

Abstract

This paper explores a model of “semantic cognition” first described in Rogers et al. (2004). This model was shown to reproduce the behaviour of neurological patients who perform poorly on a variety of tests of semantic knowledge; thus purporting to provide a comprehensive explanation for semantic deficits as found in patients with semantic dementia and, as extended in Lambon Ralph, Lowe, and Rogers (2007), individuals with herpes simplex virus encephalitis. Therefore, not only does the model emulate these semantic impairments, it also underpins a theoretical account of such memory disturbances. We report preliminary results arising from an attempted reimplementation of the Rogers et al. model. Specifically, while we were able to successfully reimplement the fully-functioning model and recreate “normal” behaviour, our attempts to replicate the behaviour of semantically impaired patients by lesioning the model were mixed. Our results suggest that while semantic impairments reminiscent of patients may arise when the Rogers et al. model is lesioned, such impairments are not a necessary consequence of the model. We discuss the implications of these apparently negative results for the Rogers et al. account of semantic cognition.

Keywords: semantic memory model; semantic dementia; backpropagation through time.

Introduction

Several connectionist models of “semantic cognition” have been developed. The goal of such models is to reproduce results obtained from testing both healthy and semantically compromised individuals on tests held to tap semantic knowledge. This is accomplished by first teaching the model to function like a healthy semantic system and then by “damaging” the model in a way that parallels the lesions seen in patients. Typically, models implement a theoretical framework that aims to explain the semantic system in both a normal and degenerate state, and evidence in support of the framework is adduced by appealing to the behaviour of the fully-functioning and lesioned model.

A complete understanding of semantic deficits, and of semantic memory in general, has not yet been reached. However sophisticated attempts to account for the deficits of some of the patient populations have been made, such as the Rogers et al. (2004) model. This model proposes that a central amodal semantic “hub” is reciprocally linked to modality-specific “spokes”, which themselves extend into so-called modal pathways. This connectivity allows fully grounded perceptual input to give rise to amodal abstract concepts. In other words, the hub itself creates semantic representations via its recurrent connections to sensory regions of the brain.

The hub model developed by Rogers et al. (2004), and then extended in Lambon Ralph et al. (2007), is one of the most complete models of human semantic deficits, boasting both an account of semantic dementia, a global semantic memory disorder, and herpes simplex virus encephalitis, a cause

of intra-semantic deficits. This paper reports results derived from an attempted reimplementation of the hub model, performed initially as a step towards extending the model and underlying theoretical framework to provide an account for additional semantic disorders. During the process of exploring this model, it became apparent that some of the results reported in Rogers et al., obtained from modelling clinical tests of semantic knowledge, were not robust. That is, the Rogers et al. results are not a necessary consequence of the model as described. This suggests that the computational-level description of the human semantic system offered by Rogers et al. is under-specified.

Semantic Cognition

The *semantic memory system* refers to a part of human long term memory consisting of a collection of abstract facts about the world. Semantic knowledge underpins linguistic meaning, providing a substrate for reasoning and inference, for categorisation, and for the creation of prototypes or exemplars. It intuitively appears that semantic memory is an abstraction or generalisation over a set of experiences collected gradually over time and organised hierarchically, as first proposed by Collins and Quillian (1969).

Semantic cognition pertains to the process by which a non- or pre-semantic percept (e.g., a drawing of a dog, or the word “dog”) gives rise to a collection of related semantic memories (e.g., dogs are furry, and have four legs) that endow the percept with meaning. The reflex-like recollection of this knowledge produces a response related to the specific concept (e.g., identifying a line-drawing by saying: “dog”). Such a reaction is only possible if the relationship between the purely perceptual stimulus and its meaning has already been instantiated in the mind (e.g., an image of a dog is linked to the phonetics of the word “dog”). This definition implies semantic cognition can be explored using tasks that require a correct interpretation, and thus response, when probed with an appropriate stimulus. Four such tasks are used by Rogers et al. (2004) to assess both their participants’ and their model’s aptitude; these are: *confrontation naming*, where an appropriate verbal name must be provided for a picture; *word-to-picture matching*, where a linguistic label must be paired with its corresponding picture from a selection that includes distractors; *sorting*, where a selection of words or pictures must be classified under hierarchical categories; and *drawing, copying and delayed copying*, where three sketches must be created, the first recreated purely from memory in response to a word, the second by direct copy from a line-drawing, and the third from memory a short time after the direct copying subtask.

Patients with dramatically low scores on such tests were

first described by Warrington (1975). Her patients, who were in their early sixties, were tested on many aspects of their cognitive functioning in order to isolate their deficit as one of pure semantics and not one of an intellectual, perceptual, or linguistic nature. The set of behavioural symptoms found, coupled with progressive bilateral neurodegeneration of the anterior temporal lobes, are characteristic of a disorder that has come to be known as *semantic dementia* (SD), a variant of frontotemporal dementia. As seen in Warrington's study and in the many more that followed, SD causes a severe impairment of semantic knowledge, with patients performing better when tested on familiar or typical items as opposed to novel or exceptional ones.

The degenerative nature of SD appears to cause patients' semantic skills to disappear in a process of akin to the reverse of learning. This, along with the characteristics of other semantic disorders, hints at some form of functionally distinct hierarchical system in which structural damage is intrinsically linked with, and gives rise to, functional deficiencies.

The Hub Model

Overview

A central claim of the hub model of Rogers et al. (2004) is that the interactions of *attractors*, which develop through learning to represent amodal concepts within semantic space, can account for both healthy and deficient semantic cognition. Attractors are stable network states that emerge following training if recurrent connectivity exists within a connectionist network. When activation is allowed to propagate throughout the trained network in a cyclic fashion, the network's state (as represented by the set of hidden and visible unit states) will converge to one such stable configuration. These stable network states exercise attractive power over a set of neighbouring network states, collectively known as their basin of attractor, such that if the network is in any of these "nearby" states it will ultimately settle to the attractor itself. These properties, according to Rogers et al., are also found in semantic memory.

To evaluate their framework Rogers et al. (2004) develop a recurrent connectionist network model. A set of stimuli is created based on statistically analysed features of common percepts (McRae & Cree, 2002), which the model is taught to auto-associate. Post-training, the model scores on tests of semantic cognition in accordance with healthy participants. After lesioning, the impaired model exhibits deficits comparable to those of SD and HSVE patients. Thus, Rogers et al. conclude that their architecture captures some level of the internal mechanisms and sophistication of the human semantic system.

Structure and Processing in the Hub Model

The recurrent connectionist network of Rogers et al. (2004) consists of one layer of 215 visible units and one layer of 64 hidden units. The latter are fully connected both to themselves and to the visible units, which are divided into three

in/output pools each consisting of: 40 name units, 64 visual feature units, and 111 verbal (61 perceptual, 32 functional, and 18 encyclopaedic) descriptor units. All units have real-valued time-varying activations with a range of $[0, 1]$ and a bias set to -2 . The hidden units, through learning, come to represent a kind of amodal semantics associated with feature patterns represented at the visible units.

As discussed above, the stimuli on which the network is trained and tested are binary patterns with co-variance that reflects statistical properties of real-world concepts. These are directly applied to the name, verbal, and visual units. Name sub-patterns are a set of binary digits, of which only one unit may be active per pattern, i.e., they are defined orthogonally. Rogers et al. (2004) argue that this labelling strategy parallels natural language in as much as, for example, the word "robin" does not in itself carry any information about the bird to which it refers. In contrast, the visual and verbal sub-patterns represent perceptual and linguistic information, and therefore must conform to predefined prototypes. Visual properties and verbal descriptors represent statements like "has a red breast", "can fly", and facts such as "is a bird" and "is living".

To produce a response given a sub-pattern the network effectively performs pattern completion. It propagates activations until it reaches a stable state in which hidden unit states do not change on successive cycles. Once the trained network has settled, its semantic state conforms to the real-valued pattern of an implicitly learned attractor, an internal configuration that is reachable due to the recurrent connectivity of the hidden units. This in turn activates the output units, thus completing the input pattern.

Training Strategy

Pattern Set The set of patterns used by Rogers et al. (2004) to train the hub model has some very particular properties. Specifically, it contains some patterns in which visual and verbal sub-patterns are mapped onto the same name. The sharing of name sub-patterns is held to be analogous to the way a chicken, a robin, and a sparrow can all be called birds, both individually and collectively. What this amounts to here is, for example, 3 nondescript birds sharing the superordinate level name "BIRD"; forming a unidirectional 3-to-1 mapping from the three pairs of visual and verbal sub-patterns to a single name label. Conversely, if given "BIRD" their network "learned to generate visual and verbal properties common to most [birds]" (Rogers et al., 2004, p. 214). Based on the statistical properties of visual and verbal co-occurrences within various categories reported by McRae and Cree (2002), Rogers et al. constructed a set of 48 patterns, with 8 patterns for each of 6 categories (mammals, birds, tools, vehicles, household objects and fruits, although only the first four have associated category-level exemplars), and 40 unique names.

In order to replicate the hub model, we constructed a statistically equivalent set of patterns, based on the probabilistic prototype for pattern creation given in fig. 3 of Rogers et al.

(2004). A comparison of the resulting dendrogram showing pattern similarity with that of fig. 2 of Rogers et al. confirms the two pattern sets are equivalent in structure.

Learning Algorithm The learning algorithm used by the original network is described only as “a variant of the backpropagation learning algorithm suited to learning in a recurrent network” (Rogers et al., 2004, p. 208). J. L. McClelland (personal communication, 2011) confirmed that this was a variant of backpropagation through time (BPTT), with the network “unrolled” (allowed to run) for 28 time-steps (Rogers et al., 2004, p. 215).

In the work reported here we adopt classic epochwise BPTT (Williams & Zipser, 1995, p. 447, eq. 18-19), with a learning rate of 0.001 and with time-averaging applied to post-synaptic unit states (McClelland, 2011). Time-averaging is a statistical method of noise reduction that may be applied over any time-varying property of a dynamic system. It has the ability to increase the signal-to-noise ratio and, in this case, results in a decrease in training epochs and for more complex mappings to be internalised, given the training details in Rogers et al. (2004)

Healthy Behaviour of Hub Model

After training for 15,000 cycles, our replication of the Rogers et al. (2004) network robustly maps names to visual and verbal sub-patterns. Thus, given a name such as “chicken”, the visual and verbal units of the network take on patterns (once the network has settled) that correspond to the visual and verbal features associated with “chicken”. Similarly, when given the visual features of that pattern, the other visible units take on values associated with the name and verbal features of the pattern. More critically, when given a superordinate name the sets of units corresponding to visual and verbal sub-patterns take on states that amount to the weighted average of the three nondescript patterns that share that same name. Conversely, when provided with the visual or verbal descriptors the network activates the general-level name. This demonstrates that the network has created stereotypes or archetypes for each category.

Semantic Tasks

Overview

We tested our network on each of the four tasks described in the introduction. In each case the method used to probe the network consists of: keeping the relevant input constant while running the network for 12 time-steps; then allowing the network to settle without any externally applied input until equilibrium is reached; and finally comparing the states of the units in the pool currently of interest to those in the relevant pattern. This is as described in Rogers et al. (2004). Following training, our network functions in its healthy state at the same general levels as Rogers et al., both in terms of training error and on all four tasks. It is therefore appropriate to consider the network’s behaviour following lesioning.

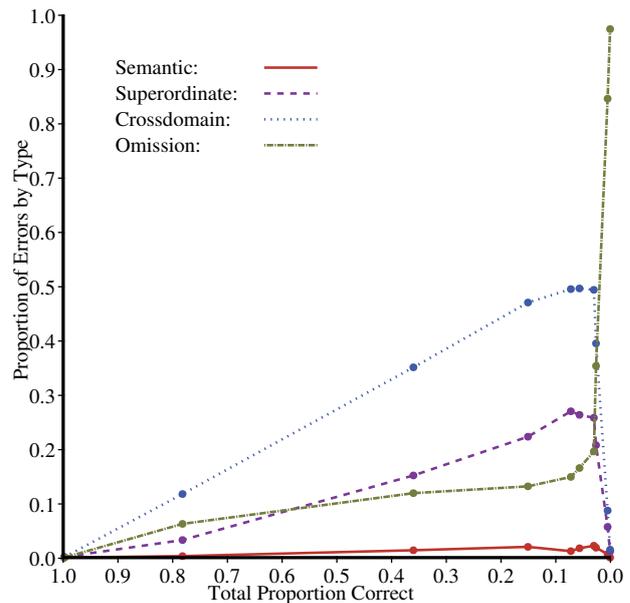


Figure 1: Results of the confrontation naming task: each data point represents the proportion of error for each type at a percentage of connections lesioned: from 0% to 90% in increments of 10%. (Compare with Rogers et al., 2004, fig. 6.)

Lesioning the Model

Rogers et al. (2004) lesioned the original hub model by indiscriminately globally severing connections between units. This zeroing of the weights is claimed to be a sufficient analogue to the damage seen in the temporal lobes of SD patients. By removing randomly selected connections in increasing percentages Rogers et al. maintain that the network displays neurodegeneration-like behaviour reflecting the decay in SD. This approach is mirrored in our replication, with semantic testing performed on three of the four standard tests described previously. Each lesion begins from a random selection of weights and is then increased, as would occur over time to an SD sufferer. For each task this is done 50 times at 10 levels of damage (i.e., from 0% to 90% of connections removed); paralleling 50 SD patients tested at 10 stages of progressive degeneration. Once the network is lesioned, settling becomes increasingly difficult and may result in dramatically different responses given the same input; thus the all results are based on sampling the current implementation 10 times for each of the sub-patterns it is tested on.

Confrontation Naming

Recall that confrontation naming requires subjects to generate verbal labels (names) from visual input (pictures). Rogers et al. (2004) report data on this task from 15 SD patients. At the earlier stages of degeneration, *omissions*, when the participant gives no answer, are relatively few but they increase dramatically as time goes by, until the only errors are omissions, i.e., the individual is completely anomic. *Superordinate er-*

rors, so called because the response is not the expected name (e.g., “owl”), but something more general (e.g., “bird”), seem to follow a similar trend to omission errors. However, at the most severe stages of the disease, superordinate errors drop off due to anomia. *Semantic errors* occur when the response is from the same category as the line-drawing presented (e.g., “dog”, when the correct answer is “horse”); these errors are low initially, then rise, and finally return to a low level (again due to anomia). *Cross-domain errors*, where a response is given from the opposing domain to that which the stimulus belongs to (e.g., calling a “horse” a “car”), are almost never documented in the SD sample.

The results of our replication of the confrontation naming task are shown in fig. 1; however, the trends shown in the behaviour of the SD patients described above and the modelling results of Rogers et al. (2004) are not shown here. In regards to our model, the largest proportion of errors from 10% to 70% of weights lesioned are cross-domain errors. This means that name units corresponding, for example, to artifacts are activated when an animal is visually presented to the network and vice versa. Omission errors are defined by Rogers et al. to occur when the network fails to activate any name unit beyond a threshold of 0.5. Changing this threshold affects the relations between the error types, but does not result in a better fit to patient data. The greater the threshold the more errors are classified as omissions, and thus the remaining three kinds of naming error (semantic, cross-domain, and superordinate) are fewer; the inverse also holds. In conclusion, the reimplementing of the hub model on the naming task does not recreate the error pattern seen in the patients.

Sorting Words and Pictures

This task requires the network to classify name and visual sub-patterns into their respective categories and domains. In fig. 2, a graph of the network’s performance at sorting at increasing levels of lesioning is shown. The scores for the two general levels of sorting (represented as solid lines), for words and for pictures, follow a descent from correct to chance levels. This is expected due to the architecture of the patterns: there are two encyclopaedic units that represent the mutually exclusive facts “is an animal” and “is an artifact”. In much the same way, the network’s scores on the two specific sorting tasks also appear to deteriorate to chance level, this time as there are 5 categories to choose from chance is at 0.2 (as in Rogers et al., 2004, fruit is excluded in the testing phase).

These results are relatively similar to those produced by the 12 patients tested by Rogers et al. (2004), however, there appears to be an important difference: the SD patients retain the ability to classify pictures into their respective domains well into their illness. Thus, while sorting into lower level categories is a skill that is largely lost, the two main semantic domains remain intact in SD; this also can be seen in fig. 8 of Rogers et al. While the original hub model appears to capture this dissociation, the current implementation does not. Arguably, the sorting of pictures is slightly more preserved than that of words, in fig. 2, but the SD patients are all at ceiling.

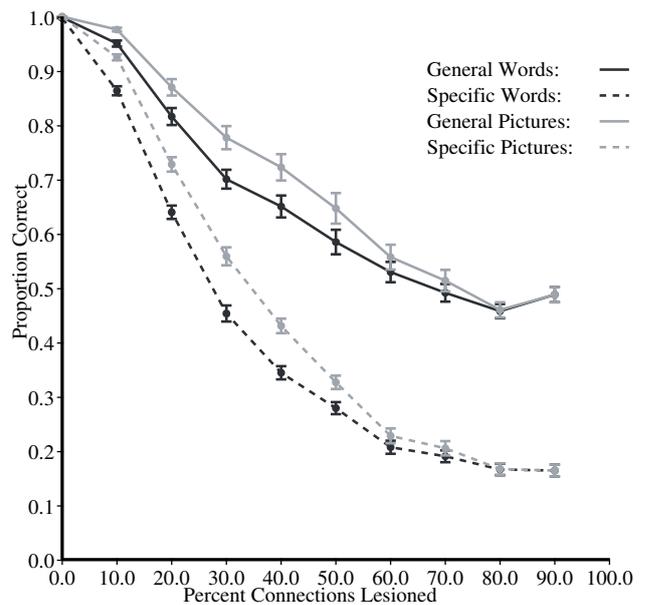


Figure 2: Results of the sorting task on words and on pictures. Error bars indicate one standard error (SE) about the mean. (Compare with Rogers et al., 2004, fig. 8.)

Again, the model is unable to fully capture this pattern of SD patient performance.

Drawing and Delayed Copying

This semantic test involves creating drawings given a name and copies based on visual sub-patterns. The results obtained from running the drawing and delayed copying semantic test on the reimplementing (see fig. 3) appear to qualitatively match those in fig. 11 of Rogers et al. (2004). Both SD patients and the model show an increase in the errors they make when drawing and copying. Also the difference between drawing and delayed copying, that the former is more difficult than the latter per patient, is reflected in both the original model and our reimplementing.

However, when the results are further analysed, as in figs. 4 and 5, a different picture emerges. Rogers et al. (2004) argue that there is an underlying distinction between the scores in each domain for two kinds of error: an *omission*, a salient feature that should have been drawn but is left out by the participant (e.g., forgetting to depict a swan with wings); and an *intrusion*, a property that perhaps holds for most exemplars but is incorrectly included in the drawing (e.g., adding four legs to a swan). In the patients’ drawings there are significantly more intrusions for animals than for artifacts (Rogers et al., 2004, p. 227), but no such effect for omissions. In fact, the original hub model only partially reproduces these effects, correctly showing more intrusions for animals but incorrectly showing more omissions for artifacts (see figs. 12-13 in Rogers et al., 2004). In our reimplementing of the hub model, we found that omission errors (both when copying and drawing) are higher in artifacts over animals (see

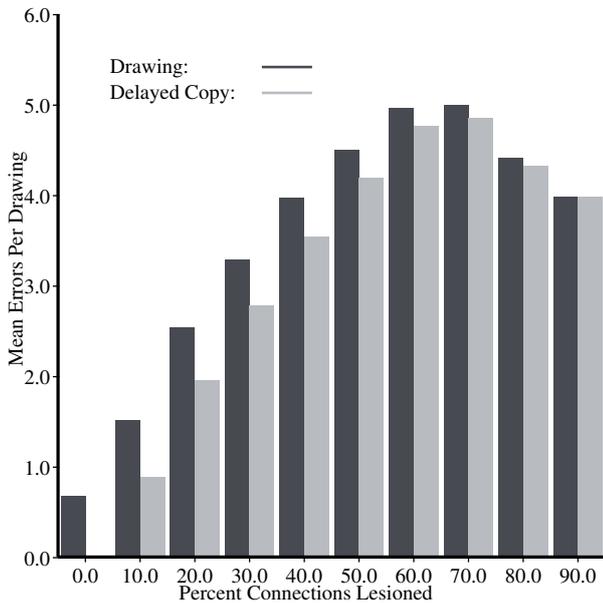


Figure 3: Mean overall feature errors per drawing for the drawing and delayed copying task for each lesioning level. Error bars not included because $SE < 0.002$. (Compare with Rogers et al., 2004, fig. 11.)

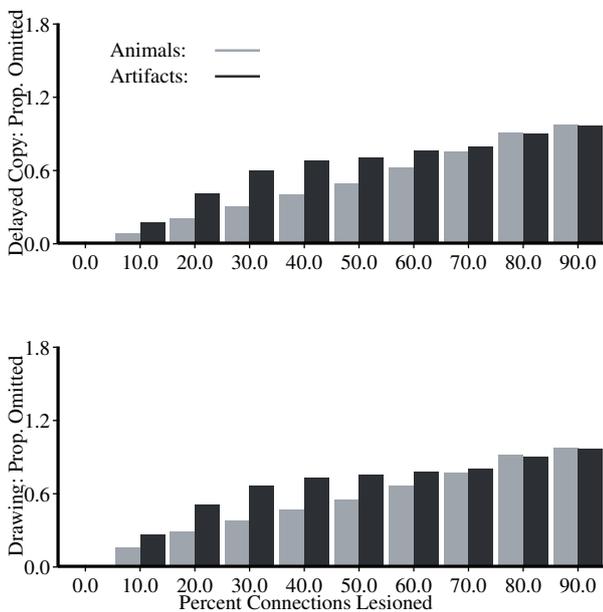


Figure 4: Proportion of errors of omission per drawing for each domain for the drawing and delayed copying task. $SE < 0.003$. (Compare with Rogers et al., 2004, fig. 12.)

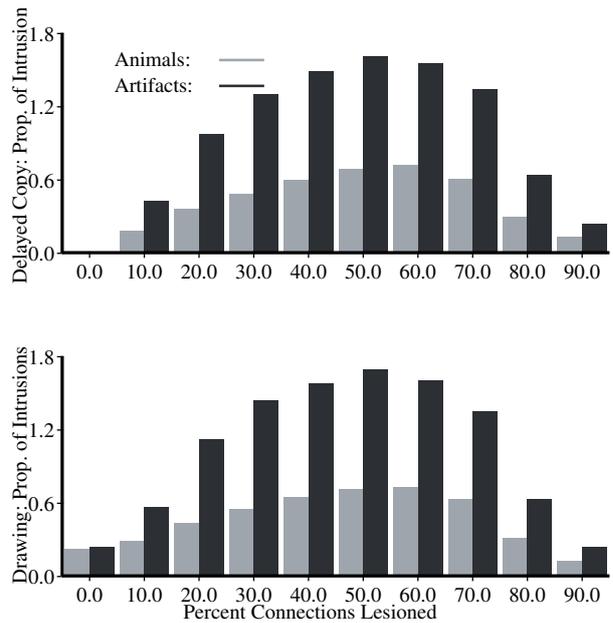


Figure 5: Proportion of intrusion errors per drawing for each domain in the drawing and delayed copying task. $SE < 0.105$. (Compare with Rogers et al., 2004, fig. 13.)

fig. 4), though with increased lesioning severity omission errors seem to occur equally in both domains (in contrast to patients, who show no effect; while the original hub model shows the same effect as our model). The rate of intrusion errors by domain reflects neither the patient data nor that of the original hub implementation, with more intrusion errors for artifacts than animals over most of the range of lesion severity (see fig. 5).

Discussion

Rogers et al. (2004) presented a model of the semantic system which they argued could account, when lesioned, for many of the deficits associated with semantic dementia. In support of this argument they report a number of simulations. We have attempted to replicate these simulations, but with mixed success. Thus, while we were able to recreate the basic learning performance of the model, we were unable to fully reproduce the patterns seen in the lesion studies.

Rogers et al. (2004) parallel the emergence of attractors with the learning of concepts, and propose that such knowledge is amodal: the somato-sensory input from the various modality-specific pathways is encapsulated by the hidden units, which thus form semantic representations. This basic theoretical notion is successfully captured by the hub model. For the case of the deficits seen in their SD patients, Rogers et al. appeal to the attractor basins' properties post-lesioning (zeroing of connection weights). They claim that animals are a tight cluster of similar concepts, thus consisting of many neighbouring attractors, while attractors for arti-

facts are distal (to the average central point of their domain), which means they form distinct conceptual loci in semantic space, and therefore their attractors are further apart. When connections are zeroed the attractor basins for living creatures are held to decay to form a larger super-attractor, which has a combined attractive power; meaning categorisation of input as an animal is possible, but access to individual features might be lost. Conversely, the attractor basins of non-living things do not merge; instead they maintain their individual attractors, albeit with distorted basins, allowing slightly better performance in this domain. The evidence put forward for the this phenomenon is the series of graphs generated from testing the Rogers et al. model. Yet the behaviour reported in the original hub model is not found in the network trained here. Why might this be so?

One possibility is that there is an error in our replication. We do not believe this to be the case, particularly given that we have simulated the basic learning performance of the network. A second is that the difference in results relates to some difference between, for example, the learning algorithm as implemented here and as implemented by Rogers et al. (2004). This is certainly possible, given that the algorithm is not fully described in the original publication. A third is that the attractors formed by the model are dependent upon the initial random weights of connections prior to learning or the order of exemplars in the training set. However, if either of these latter two situations is the case then it calls into question the theoretical explanation offered by Rogers et al. for their results.

An important aspect of this modelling strategy, that is related to the formation of attractors, is the claimed distribution of pre-semantic (perceptual and functional) features: animals and plants are closely perceptually related to each other (due to the fact they have evolved from a common ancestor and thus are composed of generally similar body parts); whereas tools, vehicles, and other inanimate objects are not similar to each other (as they have been created by humans to solve different problems, so by definition artifacts are distinct from both living things and from each other). Without training sets that encode patterns in this specific way, no connectionist model would be capable of producing a good fit to patient data. On this argument, the features, whose extraction from the environment itself is not modelled, play a pivotal role in giving rise to the semantic system's structure, and this is the case regardless of the network topology (be it recurrent or feedforward) or the learning algorithm. This is to say that, to a large extent, input to the semantic system should drive its organisation and dictate the way semantic knowledge will decay. Despite this fact, the patterns used here are unable to affect the internal structure of the reimplemented hub model in the way needed when the network is damaged. This means that the qualitative and consistent effects required post-lesioning are in fact *not* guaranteed merely by the structure of the training set. It appears that lesioning the recurrent network model by severing connections does not necessarily

result in the kind of well-behaved breakdown and generalisation of attractors as supposed by Rogers et al.

To summarise, the differences between the models appear to be due to the results obtained in Rogers et al. (2004) depending on some unarticulated implementation detail. If this is so, then the required behaviour is not a necessary consequence of the model – the original model is underspecified (perhaps our implementation of the BPTT algorithm yields attractors with different properties to the implementation of Rogers et al.). Alternatively, it may be that the behaviour of the network when damaged depends upon, for example, some apparently irrelevant factor such as the random initialisation of the connection weights. Whatever the underlying cause of the discrepancy, further investigation is needed to discover exactly why the results obtained here differ from most of those detailed in Rogers et al. If their results are in fact reproducible, but require a very specific set-up, this suggests that the model as previously reported is insufficiently specified. Conversely, if the success of the original model is due to an artefact or randomly occurring noise then this indicates that in models of this type it is critical to present results from multiple trained models, rather than from just one, to establish whether behaviours are a necessary consequence of the model or merely one of several possible outcomes.

Acknowledgements

We are grateful to Eddy J. Davelaar for constructive input during the modelling work presented here.

References

- Collins, A., & Quillian, M. (1969). Retrieval time from semantic memory. *Journal of verbal learning and verbal behavior*, 8(2), 240–247.
- Lambon Ralph, M., Lowe, C., & Rogers, T. (2007). Neural basis of category-specific semantic deficits for living things: evidence from semantic dementia, HSVE and a neural network model. *Brain*, 130, 1127–1137.
- McClelland, J. L. (2011). Explorations in Parallel Distributed Processing: A Handbook of Models, Programs, and Exercises [Computer software manual].
- McRae, K., & Cree, G. (2002). Factors underlying category-specific semantic deficits. *Category specificity in brain and mind*, 211–249.
- Rogers, T., Lambon Ralph, M., Garrard, P., Bozeat, S., McClelland, J., Hodges, J., et al. (2004). Structure and deterioration of semantic memory: a neuropsychological and computational investigation. *Psychological Review*, 111(1), 205–234.
- Warrington, E. (1975). The selective impairment of semantic memory. *Quarterly Journal of Experimental Psychology*, 27(4), 635–657.
- Williams, R., & Zipser, D. (1995). Gradient-based learning algorithms for recurrent networks and their computational complexity. In Y. Chauvin & D. Rumelhart (Eds.), *Backpropagation: Theory, architectures, and applications* (pp. 433–486).

Memory and Contextual Change in Causal Learning

Uwe Drewitz (uwe.drewitz@tu-berlin.de)

Stefan Brandenburg (stefan.brandenburg@tu-berlin.de)

Department of Cognitive Psychology and Cognitive Ergonomics, Franklinstr. 28/29
10585 Berlin, Germany

Abstract

Declarative memory is a central resource for reasoning processes. In line with the ACT-R theory, we assume that declarative memory is the basis for causal learning. Based on this assumption we conducted an experiment, showing that subjects' confidence in causal predictions decreased if their causal knowledge is discredited. Moreover, confidence decreased not only for the causal knowledge that was discredited, but also for knowledge that was not at all manipulated. Additional to the experimental results, we present an ACT-R model that perfectly fits the data and provides an explanation for the empirical findings. Contextual change turns out to sufficiently explain the empirical data and the principle of our ACT-R model.

Keywords: Contextual change, activation, causal knowledge, inference.

Introduction

The central role of memory has been investigated in a widespread range of tasks. There is much evidence showing that especially declarative memory accounts for human performance usually seen as smart or intelligent behavior (e.g. Anderson, 2007). We assume that causal learning and causal reasoning is largely based on declarative memory as well. This assumption is in line with recent research on reasoning (Mehlhorn, Taatgen, Lebiere & Krems, 2011) and the application of heuristics (e.g. Schooler & Hertwig, 2005). This research was and still is based on the ACT-R theory (Anderson, Bothell, Byrne, Douglass, Lebiere & Qin, 2004). In ACT-R human declarative memory is responsible for the storage of factual information. This information is stored in chunks. These chunks become available for retrieval, based on their activation. The higher the activation, the higher is the probability of retrieval and the faster is the retrieval of a chunk. This is the central functional principle of declarative memory in ACT-R. The activation of a chunk reflects both, the history of its usage as well as its relevance for the current context. Both aspects of activation are relevant for the explanation of human performance.

Decision-making under uncertainty is an example where human performance relies on declarative memory (Tversky & Kahnemann, 1974; Hertwig, Herzog, Schooler & Reimer, 2008, Gigerenzer & Gaismeier, 2011). Human behavior in such situations can be explained by retrieving instances of memory. However, peoples' performance cannot be explained by the mere retrieval. Instead in literature principles are proposed, which are related to the retrieval. First, in the availability heuristic (Tversky &

Kahnemann, 1973) subjects evaluate how available or how accessible (Kahnemann, 2003) a memory chunk is. Second, Schooler and Hertwig (2005) propose that people evaluate the difference in retrieval times for alternatives. This research assumes those peoples' confidences ratings in decision-making under uncertainty dates back on these by-products of the retrieval process. Also for causal learning, Drewitz and Thüring (2009) showed that peoples empirical data can be explained based on the interpretation of retrieval times. As ACT-R frames retrieval times as dependent on activation it can be concluded that confidence of ratings are directly related to the activation of memory elements. But this claim holds only for performances and experiences solely based on memory retrieval. To conclude, from the ACT-R point of view these performance and confidence ratings are explained by activation. The model results presented in this paper give evidence to this position.

Sufficiency and Necessity in Causality

It has been proposed, that human causal learning relies on cues to causality (Einhorn & Hogard, 1986). One of these cues is the co-variation between events, which people can obtain from contingency data. Theoretical approaches that emphasize the role of covariation assume that in causal learning and reasoning persons rely on frequencies of (co-) occurrence and (co-)absence of event. Figure 1 shows how this contingency information can be depicted for two events. The four cells represent the four possible pairings of two events (*C* and *E*). With respect to these two events, every observation can be assigned to one pairing and as such, to one cell of the contingency table. Moreover, every observation gives either positive or negative evidence to one of two aspects of causality: *sufficiency* and *necessity*. People are willing to attribute a causal relation between two events if both aspects are met. John Stuart Mill (1869) first made this claim.

According to him, people don't acquire causal knowledge from the repeated observation that one event follows the other. Instead they take into consideration what happens if a putative cause does not occur. From this perspective, causes can be characterized in terms of *sufficiency* and *necessity* and both of these aspects have to be satisfied. And they are satisfied to the full extent, if a number of observations fall into cell a and d as well, but not in cells c or b. In other words, every observation that belongs to the event pairing of cell a gives positive evidence to the *sufficiency* of the putative cause *C* for *E*, the effect of

interest. Just as all observations that belong to the pairing of cell d give evidence to the *necessity* of C for E.

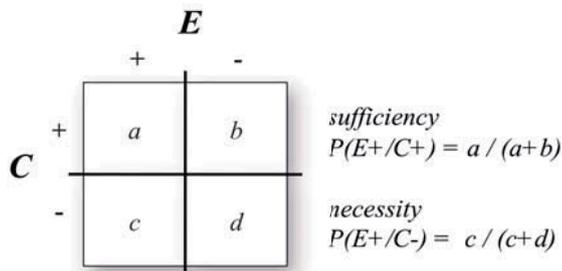


Figure 1. 2x2 contingency table ('+' indicates presence, '-' indicates absence).

Moreover, *sufficiency* and *necessity* are statistically independent of each other. Whereas the *sufficiency* of C for E depends on the frequencies in cells a and b, the *necessity* is determined by the frequencies in cells c and d (see Fig.1). Two different conditional probabilities capture these facts (see Fig. 1): the probability of the presence of E given the presence of C, $P(E+/C+)$, and the probability of the presence of E given the absence of C, $P(E+/C-)$. Positive evidence for one aspect (observations that fall either into cell a or d) can be understood as strengthening an aspect. Comparably, negative evidence (observations that fall either into cell b or c) weakens one of both aspects. Theories that emphasize the role of co-variation as cue to causality (for review see Perales and Shanks, 2007), describe how people integrate their knowledge about both aspects. That seems to be important especially when participants in a causal learning / reasoning task are requested to rate the strength of a causal relation. Of course, people do integrative judgments like that also in real-world tasks. But very often they for example make predictions based on data. In turn, as soon as people can rely on e.g. C+, their prediction should be related only to sufficiency of C for E (cells a and b). In such a case, there is no need to integrate the information about the opposite i.e. C-, which is captured by the frequencies in cells c and d. This is also true the other way around. To sum up, for the predictions based on given data, there is no need to integrate information that would hold for the absence of that data. Consequently, given the independence of both aspects, neither positive nor negative evidence related to one of the aspects should affect inferences related to the complementary aspect. Standard theories (see Perales and Shanks, 2007) do not propose such an effect. In contrast, we claim that such an effect is there. The underlying assumption is, that people do not render *sufficiency* and *necessity* as independent as they are from a mathematical point of view. Moreover, we assume, that they treat them as belonging together. And in fact, as complementary parts they belong together in terms of the concept of causality. With respect to observations people make in the world, both parts are summing up to a bigger whole – our knowledge of causal relations. But if people treat them as

parts, which shape together as a whole, it can be assumed that if one part fails, people do not longer trust in the other part. In turn our hypothesis states that the impact of negative evidence for one aspect of causality is twofold. First of all it weakens the aspect that was *discredited* by negative evidence. As a result peoples confidence for predictions related to that aspect would drop down. Thüring, Drewitz & Urbas (2006) showed this effect. Second, the complementary aspect, i.e. the aspect that is not discredited will be *devaluated*. That means peoples confidence for predictions to that aspect will decrease as well. This would be in contrast to the fact that sufficiency and necessity are independent of each other. We tested this hypothesis in our Experiment.

Basic Causal Models

With respect to the concept of causality building upon sufficiency and necessity Thüring & Jungermann (1992) proposed that people's representation of causal knowledge could be described in terms of causal models. There are different basic causal models, which can be used to represent basic or if combined, more complex causal relations.

The building blocks of all these models are conditional rules. For every causal relation the aspect of sufficiency as well as the aspect of necessity is captured by one or more of these rules. In Table 1 the sets of rules for two basic causal models are shown. These are the models that are addressed in our experiment: the *model of unique causation* and the *model of compound causation*.

Table 1. Basic causal models.

model of unique causation	model of compound causation
R1: C+ → E+	R3: C+ and X+ → E+
R2: C- → E-	R2: C- → E-
	R4: X- → E-

Experiment

In our study, participants had to acquire causal knowledge about a simulated technical system based on inductive learning. Over the course of the experiment, positive as well as negative evidence was presented to investigate the consequences of discrediting and devaluation.

Method

Participants. Fifteen graduate and undergraduate students at the Berlin Institute of Technology were recruited for the experiment. All of them were paid for their participation.

Material. Figure 2 shows the schematic screen layout of the simulated system that was presented to the participants. It was introduced as an electrical system of a power plant. The system was built up from four subsystems that were responsible for two output systems. Information about the state of these subsystems was displayed on four dials (for top boxes in Fig.2). Each dial represented the state of one

subsystem, which was either DOWN (C+) or UP (C-) or UNKNOWN because its dial was switched off. In the first of two blocks only one subsystem was causally relevant and its state served as cause (C) for the outcome (either E+ or E-) of the relevant output system (E). In the second block the same subsystems (C) together with another subsystem (X) was causally relevant. During the first block X was always set to UNKNOWN. The other two subsystems were irrelevant for the task. In both blocks they were used in some trials as distracters to give the system a more diversified appearance.

In the lower half of the screen, the displays for the output systems were shown. In some of the trials participants had to predict the outcome of only one of them and in the remaining trials they had to predict the outcome of both. If only the outcome of one system had to be predicted the display of the other output system wasn't shown. Whereas one output system (E) was relevant for the experiment the other was used to make the task more realistic. Below the display of each output system two buttons were shown for the prediction of the outcome (depicted as '+' and '-' in Fig. 2.) One button served the prediction of MALFUNCTION (E+) and the other one the prediction of OK (E-). Clicking on one of them was necessary to make the prediction. Finally, below these buttons a slider was presented (see Fig. 2) that could be adjusted to rate the confidence of the predictions. The lowest confidence (0%) was set in the middle of the slider, between the two maxima (100%), each related to one of the two possible outcomes.

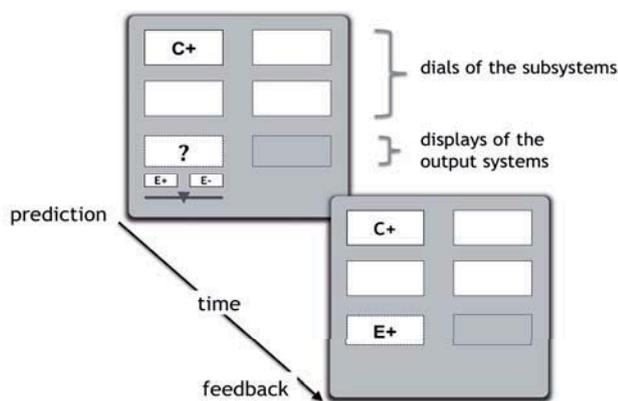


Figure 2. Screen layout (schematic) as used in the experiment for prediction and presentation of feedback.

Procedure. The participants' task was to predict the outcomes (E+ or E-) of the output system(s). To solve this task, they had to understand the underlying causal relation between the subsystems and the output systems.

In each trial, they were shown the layout of the device as presented in Figure 2. First, subjects had to check the operation of the subsystems. Based on this information, they were requested to predict the state of the output system(s) by clicking on the respective buttons (OK or MALFUNCTION). Finally, they rated their confidence for

each prediction by adjusting the respective slider(s). After participants finished their prediction and confidence rating, they had to click on a 'send' button and subsequently received feedback that showed the actual outcome(s).

The experiment consisted of two blocks, each of them with a *learning phase* and a *test phase* as shown in Figure 3. The blocks differed in the complexity of the underlying causal relations and the number of trials in the learning phase. During the learning phase positive evidence for one type of causal relation was provided. In the learning phase of the first block participants received information that enabled them to acquire a model of unique causation with the two rules R1 and R2 (see Tab. 1). In the learning phase of the second block subjects received information, which supported the acquisition of the two new rules R3 and R4 (see Tab.1). Thus subjects could learn a new model, the model of compound causation (see Tab.1).

Additionally, we presented distracter trials with information about the irrelevant subsystems and trials were participants had to predict the outcome of the second output system that was irrelevant for the test of the hypothesis. Relevant for the test of the hypothesis were the pre-measure and the post measure in each block. For both blocks, the last trial of the learning phase served as *pre-measure* (see Fig. 3). In the respective trial in block one, people had to make a prediction based on C- and received as feedback E-. In the respective trial for block two we presented C+ and X+ and gave people after they made their prediction the feedback E+. These pre-measure trials served as positive evidence too. That's why for both blocks in Figure 3, the number of one cell increases from the learning to the test phase.

Subsequently to the learning phase, the test phase started. In four of these trials, we presented negative evidence (see Fig. 3) for one aspect of causality. The negative evidence was given with respect to the causal relations that were supported before. You can see the number of presentations of negative evidence in the black boxes in Figure 3. In block one, we showed four times C+, E- and in block two we presented two times C-, E+ and two times X-, E+ (together four times negative evidence). Again, distracter trials and trials focusing on the irrelevant output system were presented. In the last trial of the test phase, the *post-measure* was recorded (see Fig.3). To accomplish this, the same data were given as in the pre-measure trial. For block one that was C- and for block two C+ and X+.

Independent and dependent variables. To investigate the *strengthening* of rules, the amount of *positive evidence* ranged from one to sixteen trials (see Fig. 3, positive evidence) for the respective rules (R1, R2, R3 and R4). To test the impact of discrediting, the amount of *negative evidence* ranged from one trial to four trials (see Fig.3, negative evidence) for the respective rules (R1, R2 & R4).

The factor *measurement* with the factor levels *pre* and *post* served the investigation of devaluation as described in the procedure (see Fig.3). Throughout the experiment,

confidence ratings of inferences predicting the states of the relevant output system were used as dependent variable.

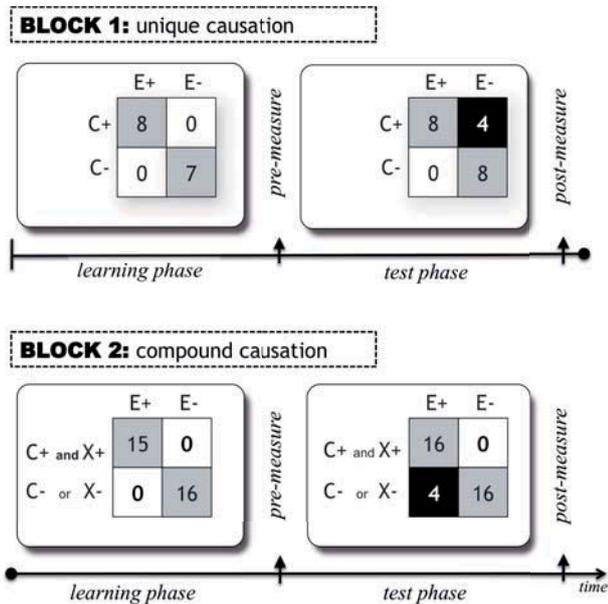


Figure 3. Experimental procedure (schematic). Contingency tables for both blocks display frequencies of positive evidence (grey) for the learning phase and negative evidence (black) for the discrediting phase. The last presentation of positive evidence served as pre-measure. The contingency tables on the right display summed frequencies for positive and negative evidence.

Results

For statistical analysis, we computed three ANOVAs with repeated measures, one for each effect. Additional to the significance of effects we report the effect size f after Cohen (1988). However, *strengthening* greatly affected subjects confidence ratings over the course of the learning phases, $F(7,98)=11.47$, $p<0.01$, $f=0.91$. Therefore, subjects' confidence in their prediction of the state of the output system strongly increased over time. Additionally, we obtained an effect for rule, $F(3,42)=7.46$, $p<0.01$, $f=0.73$. Hence, it was easier for subjects to acquire basic rules (R1 & R2) compared to the more complex rules (R3 & R4).

For *discrediting*, we found a significant large main effect of *negative evidence* over time, $F(1,14)=4.21$, $p=0.05$, $f=0.54$. Hence, subjects showed lower confidence in their ratings about rules that were discredited.

To investigate the effect of *devaluating* a rule, it seems necessary to highlight how we achieved the data for this computation. For all subjects R1 and R2 (in the first block) and R3 and R4 (in the second block) were strengthened. The last trial of R2 in the learning phase of block 1 and R3 in the learning phase of block 2 served as pre-measure for the devaluation. However, only R1 was discredited in the first block and R2/R4 were discredited in the second block.

If subjects' confidence for the prediction of R2 in the first block (post-measure) and R3 in the second block (post-measure) was lower after discrediting the other rules, devaluation took place. The ANOVA revealed a medium main effect of *rule* ($F(1,28)=5.42$, $p=0.03$, $f=0.43$) and a large main effect of *measurement*, $F(1,28)=19.40$, $p<0.01$, $f=0.83$). Therefore subjects' confidence was lower for R2 compared to R3. Additionally, devaluation lead to significantly lower confidence for participants' confidence for both rules (R1 & R3) after R1, R2 and R4 were discredited.

Model Description

In order to examine, whether the empirical observed effect of devaluation could be explained based on declarative memory and the concept of activation, we set up a simple ACT-R model. Two central assumptions were made to specify the model. At first, we assumed that the task could be processed solely based on instance retrieval.

The second assumption we made was in contrast to the theoretical considerations, which guided the formation of the hypothesis of the devaluation effect. To model the effect found in the empirical data, we assumed that negative evidence, which per definition contradicts observations made beforehand, would be considered as a contextual change (see Block and Reed, 1978). Accordingly, people are aware of changes in the context internally (cognitive context) as well as externally (external context). Assuming that observations, which people make, and certain knowledge that they acquire accordingly, is related to a certain context would result in a change of availability of that knowledge if the context changes.

Consequently, the behavior of the model can be explained based on contextual changes and instance retrieval, i.e. the standard ACT-R 6 activation equations:

$$A_i = B_i + \sum_j W_j S_{ji} + \sum_l PM_{li} \quad (\text{activation equation})$$

Accordingly, the Activation (A_i) of a chunk i is defined by its base-level activation (B_i), the amount of activation that spreads out from a source (representation of the stimuli and the current context) and the partial matching component. W_j reflects the attentional weighting allocated to every element j on the source of activation. S_{ji} is the strength of the associative connections between these elements and the chunk i . P is the mismatch penalty and M_{li} is the similarity between the elements l specified for a request of retrieval from declarative memory and the respective elements of chunk i . The base-level B_i itself is defined as

$$B_i = \ln\left(\sum_{j=1}^n t_j^{-d}\right) \quad (\text{base-level learning equation})$$

where n is the number of presentations, t is the time that passed since the j th presentation and d is the rate of decay. Last but not least the associative strength is defined as

$$S_{ji} = S - \ln(fan_j) \quad (\text{associative strength equation})$$

where S is the maximum associative strength and fan_j the number of chunks in memory, the chunk j as an element on the source of activation is associated with, plus one for association with itself.

Model Settings. Based on these equations the activation of that chunks was calculated, which would match the retrieval request in the trials of the relevant measures. Therefore we calculated the number of presentations (see base-level learning equation), assuming that there was one encoding of screen information as well as one declarative retrieval per trial. For simplicity reasons, the respective times (t_i) were calculated based on the assumptions that all trials were processed in the same time. To determine the associative strength between the current context and the different memory chunks (representing different trials) we counted the number of distinct stimuli used in the experiment. The respective number of associative connections was assigned to each context and so the fan for each context was set.

Since there is no default, one a single parameter, the mismatch penalty, was set to 0.5. Except this, all other parameters used (see equations), were set to their defaults prescribed by the ACT-R theory (cf. <http://act-r.psy.cmu.edu/>).

Subsequently, we transformed the resulting activation values (A_i) into values representing confidence ($confidence_i$) using the following equation:

$$confidence_i = \ln(A_i) \times SF \quad (\text{transformation equation})$$

The parameter SF is a scaling factor and was set to 100. The results produced by the model are shown together with the empirical data in Figure 4. Without adjusting any parameter the model produces an excellent fit to the empirical data ($R^2=1$). The qualitative match between the model results and the human data is perfect. Quantitatively there is a clear deviation. But this deviation is of same for all conditions.

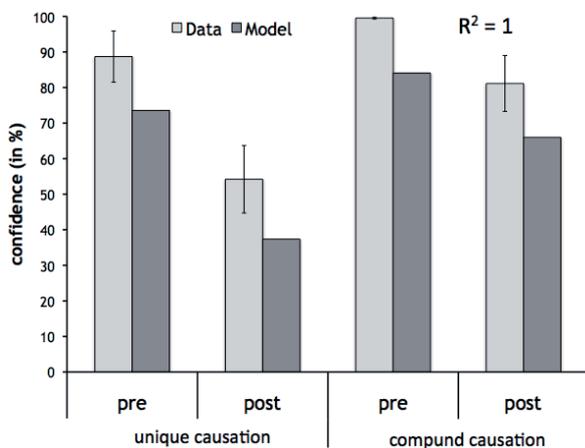


Figure 4. Empirical data and model results. Error bars represent standard error.

Discussion

The present paper investigated two effects that influence causal learning, memory and contextual change. In particular, we looked at persons' confidence ratings with respect to their predictions of certain outcomes after their causal knowledge was (a) strengthened and (b) discredited.

In line with previous research (Thüring et al., 2006), we found that the presentation of positive evidence leads to higher confidence in subjects' predictions of the state of the output system. Additionally, we replicated the effect of discrediting. Hence participants' confidence in their prediction of the state of the output system decreased, when the rules they had learned were discredited. In contrast to these well-established effects, we demonstrated the effect of devaluation. Therefore, participants' over-all confidence in their predictions of the system state decreased also when the rules they could use for prediction on given date were not discredited. This effect opposes the assumption that people treat sufficiency and necessity as independent as they are from a mathematical perspective. At least for the case of negative evidence the data support this position.

Excluding the striking effect of devaluation, memory effects (strengthening and discrediting) on confidence in causal learning were proposed and modeled in ACT-R before (e.g. Drewitz & Thüring, 2009). Extending this research, the present ACT-R model accounts for the effect of devaluation as well. For both causal relations (unique and compound causation), model data perfectly fitted to subjects' behavior in the experiment. It is important to note that this simple ACT-R model mimics empirical data about subjective confidences without any additional parameters. Hence the present data (and model) strongly support the theoretical claim we made at the beginning of this paper. There, we proposed that peoples' confidences of ratings under uncertainty are directly related to the activation of memory elements. The results presented in this research undermine this claim at least for performances and experiences, which are solely based on memory retrieval.

However, the presented ACT-R model does not only mimic the empirical data. Its working principle also provides an elegant theoretical way to explain the data. We used the concept of contextual change (Block, 1978) as basis for the ACT-R model. Contextual change means that with respect to their model or rule like causal knowledge people consider certain contexts. In our experiment (and model), subjects learned causal relations in each block. After a couple of trials, their rule-based knowledge about the functioning of the technical operating system was almost perfect. This phase of strengthening contained only positive evidence. Therefore subjects' causal knowledge about the inner workings of the technical system was enhanced. In the ACT-R model we encoded this strengthening phase as the context in which subjects acquired and used their causal knowledge. However, each time this initial strengthening phase was followed by a short phase with negative evidence (discrediting phase). In

this phase participants questioned their causal knowledge and the confidence in their predictions decreased. Thus, in the experiment we changed the experimental stimuli from strengthening causal relations to discrediting the very same. Psychologically, this change might have appeared as a contextual change. Suddenly the working principle of the technical system changed. This change was not visible to the participants, except that their learned causal relations did not lead to successful prediction of the state of the output system. Again, psychologically participants might have represented this change as a step from one working principle to another and so considered a new context.

For theories of causal learning the presented model raises some questions. From their perspectives peoples judgments and confidence ratings are seen as the result of reasoning or data integration processes. Since our simple memory model modeled the decrease in participants' confidence ratings perfectly, these more complex standard views of seem questionable. As introduced in the beginning, standard theories assume much more than memory retrieval. There are much less approaches that assume, as we do that lower ratings in causal reasoning might reflect a reduced availability i.e. accessibility of memory information due to less activation. In our model this goes back to less relevance of memory information as soon as new contexts are considered. Thus 'deliberate' causal behavior can be explained simply based on memory activation.

Of course, further experiments should replicate the present work. Additionally, it should first be tested whether our contextual change (ACT-R) model holds in other situations of causal learning as well. Second, it has to be proven for more cases that confidence ratings can be drawn from activation. The non-linear transformation function that was used has to be tested. It assumes that the higher the activation the less are changes of that activation reflected in confidence ratings drawn from it, even if the base-level learning that generates these activations already shows this kind of non-linearity.

Our next step is to elaborate more on the devaluation effect and the role of memory in causal learning and reasoning. For example if this effect occurs also for more complex causal knowledge or if similar effects can be found in reaction time data too. The presented ACT-R model will have to prove his validity for those data as well.

References

- Anderson, J. R. (2007) *How Can the Human Mind Occur in the Physical Universe?* New York: Oxford University Press.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review* 111, (4). 1036-1060.
- Block, R. A., & Reed, M. A. (1978). Remembered duration: Evidence for a contextual-change hypothesis. *Journal of Experimental Psychology: Human Learning and Memory*, 4, 656-665.

- Drewitz, U. & Thüring, M. (2009). Modeling the Confidence of Predictions: A Time Based Approach. In A. Howes, D. Peebles, R. Cooper (Eds.), 9th International Conference on Cognitive Modeling – ICCM2009, Manchester, UK.
- Einhorn, H. J. & Hogarth, R. M. (1982). Prediction, diagnosis, and causal thinking in forecasting. *Journal of Forecasting*, 1, 23-36.
- Gigerenzer, G., & Gaissmaier, W. (2011). Heuristic decision making. *Annual Review of Psychology*, 62, 451–482.
- Hertwig, R., Herzog, S. M., Schooler, L. J. & Reimer, T. (2008). Fluency heuristic: A model of how the mind exploits a by-product of information retrieval. *Journal of Experimental Psychology - Learning, Memory, and Cognition*, 34(5), 1191-1206.
- Kahneman, D. (2003). "A perspective on judgment and choice: Mapping bounded rationality". *American Psychologist* 58 (9): 697–720.
- Mehlhorn, K., Taatgen, N.A., Lebiere, C., Krems, J.F. (2011). Memory Activation and the Availability of Explanations in Sequential Diagnostic Reasoning. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 37, 1391-1411.
- Perales, J. C., & Shanks, D. R. (2007). Models of covariation -based causal judgment: A review and synthesis. *Psychonomic Bulletin & Review*, 14, 577-596.
- Thüring, M., Drewitz, U., & Urbas, L. (2006). Inductive Learning, Uncertainty and the Acquisition of Causal Models. In R. Sun & N. Miyake (Eds.), *Proceedings of the 28th Annual Cognitive Science Society*. NJ: LEA.
- Thüring, M. & Jungermann, H. (1992). Who will catch the Nagami Fever? Causal inferences and probability judgment in mental models of diseases. In D. A. Evans & V. L. Patel (Eds.), *Advanced models of cognition for medical training and practice* (pp. 307-325). Berlin: Springer.
- Tversky, A. & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science*, 185, 1124-1131.
- Schooler, L. J. & Hertwig, R. (2005). How Forgetting Aids Heuristic Inference. *Psychological Review*, 112(3), 610-628.

Acknowledgement:

We thank Niels Taatgen and Hedderik van Rijn who initially had the idea to implement the concept of contextual changes.

Functional Modeling of Personality Properties Based on Motivational Traits

Joscha Bach (joscha.bach@hu-berlin.de)

Berlin School of Mind and Brain, Humboldt University of Berlin, Unter den Linden 6
10199 Berlin, Germany

Keywords: Motivation; Five Factor Model; Personality Properties; MicroPsi; Psi Theory; Cognitive Modeling.

Motivation in the Cognitive Architecture MicroPsi

The cognitive modeling of personality traits—as exemplified by the well-known Five Factor Model (Digman, 1990; Goldberg, 1993)—requires the identification and suitable functional abstraction of underlying mechanisms within a cognitive architecture. We propose that these mechanisms are predominantly *motivational*, and are using the cognitive architecture *MicroPsi* (Bach 2003, 2009) for analysis and modeling.

MicroPsi's motivational system can be characterized by a (pre-defined) set of demands of the agent, which are represented as urge signals. Changes in these signals determine valences: a change of a demand towards its target value creates a positive reinforcement (pleasure signal), while a negative change away from the target results in a negative reinforcement (displeasure signal). These signals can be used to create associations between the urges and situations that satisfy them (goals) or frustrate them (aversive situations). In accordance with the Psi theory (Dörner 1999), MicroPsi uses three groups of demands: physiological, social and cognitive.

The *physiological demands* (food, water, physical integrity/pain avoidance etc.) become active whenever the autonomous regulation of physiological parameters fails and provide for the basic survival. Here, survival itself is seen as an abstract concept and not a demand itself.

Social demands consist in a need for affiliation with others, and are mediated by social signals ('legitimacy signals'), such as displays of affection, acceptance, rejection or reproach. The affiliation mechanism allows to structure social interaction beyond rational utility: purely social rewards are often sufficient to motivate an agent for cooperative behavior, without incurring the need to supply a material gratification and thereby affect the fitness of the group, or to discourage anti-social behavior without decreasing the agent's material fitness by doling out punishment. A second social demand is called 'internal legitimacy': it corresponds to internal social signals that are related to the conformance to internalized social norms ('honor'). Obviously, the list of social demands addressed in MicroPsi is incomplete; for instance, it lacks sexual needs (libido).

The group of cognitive demands spans needs for competence, a need for uncertainty reduction, and needs for aesthetics.

Competence is either epistemic (related to skills): it provides an estimate on the agent's ability to cope with any specific task, by delivering a reward on its successful completion, and a penalty on failures. Thus, skill-acquisition can become a goal on its own. Furthermore, competence may be general, i.e. related to the overall ability of the agent to cope with the environment. General competence delivers a heuristics on the amount of risk an agent should take, and is measured as a floating average over successes and failures of the agent's past actions.

Uncertainty reduction is aimed at discovering the outcomes of actions, and exploring the structure of objects and situations. Uncertainty reduction is satisfied by 'certainty events': the complete identification of an object, scene or frame, by fulfilled expectations (even negative ones), and by a long and non-branching expectation horizon. Conversely, uncertainty reduction is frustrated whenever the agent encounters unknown objects or events, discovers elements without a known connection to behavior, etc.

Uncertainty signals are weighted with the motivational relevance of their object. Generally, a high uncertainty will give rise to explorative behaviors, unless the agent has a low epistemic competence for exploration.

Aesthetics is a demand that directs the agent at seeking order, i.e. better representations (abstract aesthetics), or seeking out particular stimuli, based on evolutionary preferences, such as certain body schemas or landscapes (stimulus oriented aesthetics).

Each demand is characterized by several parameters:

- The *target value* v_d of the demand d
- The *deviation* $|v_d - c_d|$ from that value, represented by an urge indicator $urge_d$,
- The *weight* of the demand (its relative importance, compared to other demands with the same urgency) w_d ,
- The *gain* (the satisfaction derived from a positive stimulus or consumption) g_d ,
- The *loss* (the penalty incurred from a negative stimulus or a frustration) l_d ,
- The *decay* (the autonomous increase of the deviation from the target value over time) f_d .

Even if no gain or loss is incurred, the decay ensures that the motivational parameters change relentlessly, and the agent is requiring to constantly replenish the demands. (For a detailed description, see Bach 2011).

Application for Modeling Personality Traits

The motivational traits of agents can be defined as a set of physiological, social and cognitive demands D , each of

them annotated by a tuple (w_d, g_d, l_d, f_d) , describing the weight, gain, loss and decay of the respective demand. Using these parameters, it is possible to create agent models that conform to the Five Factor Model (FFM, or “Big Five”) established in personality psychology. The FFM suggests five dimensions of personality traits, which together can be used to characterize emotional/motivational dispositions of an individual:

- *Openness*: This describes the interest a subject takes in new situations, ideas and stimuli. Openness is associated with intellectual curiosity, appreciation of art, and non-conservatism
- *Conscientiousness*: This characterizes how organized/rigid a subject tends to be. Conscientious individuals tend to spend more time planning, attend carefully to details and attempt to follow plans and rules rigorously.
- *Extraversion*: This relates to the interest individuals take in interpersonal interaction, their surgency and expressiveness.
- *Agreeableness*: Individuals that are highly agreeable tend to avoid conflicts, are friendly and seek positive social interaction.
- *Neuroticism*: This amounts to emotional instability. Subjects with a high degree of neuroticism tend to experience negative emotions more strongly, are prone to anxiety and mood switches.

Modeling configurations of personality traits by choosing appropriate settings for the tuples (w_d, g_d, l_d, f_d) is straightforward. Since all of them are related to social and cognitive pre-dispositions, it is sufficient to look at the demands for *affiliation*, *competence*, *certainty* (= uncertainty reduction) and *aesthetics*.

For instance, a high degree of neuroticism can be expressed by choosing particularly high values for the loss and decay of *competence* and *certainty* (and possibly the other demands, too). In other words, the agent needs to replenish its competence and certainty very often, and it will react disproportionately to failures of doing so, and to frustrations of these demands. The continuous decay of *certainty* makes the agent prone to episodes of anxiety.

Conversely, an agent with the opposite settings, i.e., very low decays and losses on *competence* and *certainty* will not take a big hit on failure, and won’t need to seek out new competence and certainty rewards as often. Thus, it will display a greater degree of emotional stability and complacency (= low neuroticism).

A highly *open* agent can be modeled by a high decay on *competence* and *certainty*, too, so the agent is forced to seek out competence and exploration rewards. On the other hand, it should receive a high gain on satisfying its cognitive (and possibly social) demands. Thus, it will receive positive frequent and strong positive reinforcements of its explorative and competence building behaviors, resulting in a high tendency to seek out new situations and stimuli.

Our model determines *conscientiousness* with a strong loss factor of *competence* and *certainty*, combined with a

weak gain of *competence/certainty*. This means that the reward for exploration and skill acquisition is low, compared from the loss incurred by risking them. A high decay on *competence*, but low decay on the other drives can additionally result in a low interest in seeking out new social, aesthetic or exploratory challenges, while focusing on a high accuracy in the execution of plans and skills.

Extraversion is produced by a high decay of the *affiliation* demand, which therefore requires constant social interaction to be replenished. Strong gains on *affiliation* and *competence*, as opposed to weak losses on these drives result in a strong reinforcements due to social and competence successes, but only little aversion due to failures.

Agreeable agents are somewhat similar to extroverts due to a high decay on *affiliation* (and possibly *competence*), so they need to seek out social situations often. Unlike extroverts, they receive strong *affiliation* losses due to negative social signals, and gain little *competence*. Thus, they are likely to avoid arguments: they have little positive rewards to gain from them, but incur strong negative reinforcements if they do not succeed socially.

Currently, our model is restricted to simple multi-agent simulations. At the moment, we are using our model to design a series of problem solving scenarios that correlate personality properties with the performance of subjects (Greiff & Funke, 2009). As a result, we hope to provide a direct application of the model for psychometric purposes. Furthermore, well-defined problem solving scenarios present an opportunity to compare the performance of human subjects directly with that of computational agents and will thereby allow us to improve the motivational and emotional framework of our cognitive model.

References

- Bach, J. (2003). *The MicroPsi Agent Architecture*. Proceedings of ICCM-5, International Conference on Cognitive Modeling, Bamberg, Germany, 15-20
- Bach, J. (2009). Principles of Synthetic Intelligence. Psi, an architecture of motivated cognition. Oxford University Press.
- Bach, J. (2011). *A Motivational System for Cognitive AI*. In Schmidhuber, J., Thorisson, K. R., & Looks, M. (eds.): Proceedings of Fourth Conference on Artificial General Intelligence, Mountain View, CA. 232-242.
- Digman, J. M. (1990). *Personality structure: Emergence of the five-factor model*. Annual Review of Psychology 41: 417–440.
- Dörner, D. (1999). *Bauplan für eine Seele*. Reinbeck: Rowohlt.
- Goldberg, L. R. (1993). *The structure of phenotypic personality traits*. American Psychologist 48 (1): 26–34.
- Greiff, S., & Funke, J. (2009). *Measuring Complex Problem Solving - The MicroDYN approach*. In F. Scheuermann (ed.), The Transition to Computer-Based Assessment - Lessons learned from large-scale surveys and implications for testing. Luxembourg: Office for Official Publications of the European Communities.

Model-Based Prediction of Between-Trial Fluctuations in Response Caution From EEG Data

Udo Boehm (u.bohm.1@student.rug.nl)¹
Leendert van Maanen (l.vanmaanen@uva.nl)²
Birte Forstmann (bforstmann@gmail.com)²
Hedderik van Rijn (hedderik@van-rijn.org)¹

¹Department of Experimental Psychology, University of Groningen, Grote Kruisstraat 2/1
Groningen, 9712 TS, the Netherlands

²Cognitive Science Center Amsterdam, University of Amsterdam, Nieuwe Achtergracht 129
Amsterdam, 1018 WS, the Netherlands

Keywords: contingent negative variation; linear ballistic accumulation; speed-accuracy tradeoff.

The Role of the Pre-SMA in Decision Making

Recent models of decision making under time constraints assume that the pre-supplementary motor area (pre-SMA) modulates the excitability of an action selection mechanism implemented by the basal ganglia (e.g. Forstmann et al., 2008). The basal ganglia exert a tonic inhibition on the cortex. By decreasing this inhibition the pre-SMA can decrease response caution, thus facilitating speeded but possibly faulty responses. This claim is supported by a series of neuroimaging studies on random dot kinematograms where participants were instructed either to be as quick or as accurate as possible (Van Maanen et al., 2011; Forstmann et al., 2008). The data were analysed by fitting a linear ballistic accumulation model (LBA, Brown & Heathcote, 2008) to the decision time data and correlating the model's response caution parameter with hemodynamic response in the pre-SMA.

Forstmann et al. (2008) showed that differences in response caution between conditions with speed instructions and conditions with accuracy instructions estimated with the LBA model correlated with individual differences in BOLD response change between conditions. Van Maanen et al. (2011) applied the LBA model to single trial data. They found that trial-by-trial fluctuations in response caution under speed stress but not under accuracy instructions correlated with the single-trial BOLD response in the pre-SMA.

The Pre-SMA in EEG Data

A number of EEG studies have linked the contingent negative variation (CNV), an often-studied slow negative potential, to brain regions in close proximity to the pre-SMA and to measures that express the ease with which participants can trigger a response. Leuthold and Jentsch (2001), applying dipole source localisation to a response precueing task, found that the CNV preceding a response originates from sources close to the SMA. Moreover, a number of studies have reported a negative correlation between CNV amplitude and reaction time (e.g. Hillyard,

1968). Elbert (1990) suggested that the CNV might reflect adjustments of cortical excitability. He supports this claim with data from a signal detection experiment in which high CNV amplitudes correlated with an increase in false alarms and low CNV amplitudes increased the number of misses.

These results suggest that the CNV might reflect the same processes involved in the adjustment of response caution as the activity of the pre-SMA in fMRI studies. If this is the case, lower response caution should be observed for higher CNV amplitudes under speed but not under accuracy instructions. To further investigate this possibility we ran an EEG experiment using a random dot kinematogram and correlated the CNV amplitude with single-trial estimates of response caution from an LBA model.

EEG Experiment and LBA Modelling

Experiment

A group of 14 undergraduate students (10 female) participated in the experiment for partial course credit. They performed 200 trials of a random dot kinematogram task. EEG data were recorded from 32 scalp sites. Trials with an amplitude exceeding $\pm 250\mu\text{V}$ and trials with artefacts were excluded. Eye blink artefacts were corrected using independent component analysis. Data were low-pass filtered at 35 Hz and baseline-corrected to a baseline-window from 300ms to 100ms before the onset of the fixation cross (see below). All further analyses were based on the FCz electrode.

Participants were asked to decide whether a cloud of 120 pseudo-randomly moving dots was moving to the left or to the right. At the beginning of each trial they were instructed to either react as quickly (SP for speed) or as accurately (AC for accurate) as possible.

Each trial started with a blank screen, followed by the speed instruction and another blank screen. A fixation cross was presented for before the onset of the dot kinematogram. This was followed by a blank screen and feedback on either the response speed in the SP condition or the accuracy in the AC condition.

The CNV was measured during the presentation of the fixation cross before the onset of the dot kinematogram.

CNV amplitude was defined as the mean amplitude between 200ms and 100ms before the cloud of dots was presented.

LBA model

The LBA model describes decisions as an evidence accumulation process with two accumulators, one for correct and one for incorrect responses. Starting from an initial amount of evidence, evidence is accumulated until one of the accumulators reaches a threshold at which point a decision is made. The model includes 5 parameters. The drift rate d for the evidence accumulation is sampled from a normal distribution with mean ν and standard deviation s . The initial amount of evidence, reflecting response caution, is sampled from a uniform distribution from 0 to A . The more evidence is initially available, the less evidence needs to be accumulated and the quicker a response can be made. The response threshold b describes the amount of evidence that is needed to make a decision. Finally, the non-decision time t_0 reflects all processes not related to the decision process, such as the execution of a motor response.

The best fitting model was selected based on formal model comparisons using Bayesian Information Criterion. The selected model was one in which the mean drift rate, the standard deviation of the drift rate and the response threshold were free to vary between speed instructions. The model was fit to participants' reaction time distributions as described in Donkin, Brown, and Heathcote (2011). Subsequently maximum likelihood estimates of the single trial drift rate d and initial evidence a were obtained as in Van Maanen et al. (2011).

Results and Discussion

Linear mixed effects models were used to assess the relationship between CNV amplitude and single-trial response caution and drift rate. The first model included fixed effects for speed instruction (2 levels: AC and SP), single-trial response caution and the interaction of the two as well as a random intercept per subject. While response caution did not predict CNV amplitude in the AC condition ($\beta = -0.01, p = .25$), the significant interaction term showed it to be a significant predictor in the SP condition ($\beta = 0.04, p < .01$). To test whether drift rate explains additional variance in the CNV amplitude, we constructed a second model that included single-trial drift rate and its interaction with speed instruction as additional predictors. Comparing this model to our first model showed that drift rate did not improve prediction ($\chi^2(2) = 1.95, p = .38$).

These results imply that while participants decrease their response caution when prompted to react as quickly as possible, no such adjustment is made if accurate responding is stressed. It aligns well with the findings of Van Maanen et al. (2011) as well as suggestions that the CNV might reflect response preparation processes (Elbert, 1990). Moreover, the finding that drift rates are not related to CNV amplitude shows that the LBA model recovers the differential contribution of drift rates and initial evidence to the accumulation process.

These findings bear interesting implications for two fields of research. On the one hand, the CNV might offer a more direct measure of response caution. Instead of having to rely on parameter estimates from a model that was fit to noisy reaction time data, the CNV might provide an easy-to-obtain measure of the neuronal activity underlying response caution. On the other hand, these findings might also help resolve a long-standing debate about the role of the CNV in time estimation. Macar, Vidal, and Casini (1999) suggested that the CNV reflects the accumulation of pulses from an internal clock. However, Van Rijn, Kononowicz, Meck, Ng, and Penney (2011) argue that the CNV reflects the response preparation or decision processes. The current findings support the latter interpretation. While participants are waiting for a time interval to pass their pre-SMA might become active to prepare the selection and execution of a response, which is reflected by a higher CNV amplitude.

References

- Brown, S. D. & Heathcote, A. (2008). The simplest complete model of choice response time: Linear ballistic accumulation. *Cognitive Psychology*, *57*, 153-178.
- Donkin, C., Brown, S., & Heathcote, A. (2011). Drawing conclusions from choice response time models: A tutorial using the linear ballistic accumulator. *Journal of Mathematical Psychology*, *55*, 140-151.
- Elbert, T. (1990). Slow cortical potentials reflect the regulation of cortical excitability. In W.C. McCallum (Ed.), *Proceedings of a NATO Advanced Research Workshop on Slow Potential Change in the Human Brain* (pp. 235-251). Il Ciocco, Italy.
- Forstmann, B. U., Dutilh, G., Brown, S., Neumann, J., Von Cramon, D. Y., Ridderinkhof, K. R., & Wagenmakers, E.-J. (2008). Striatum and pre-SMA facilitate decision-making under time pressure. *Proceedings of the National Academy of Sciences*, *105*(45), 17538-17542.
- Hillyard, S. A. (1968). Relationships between the contingent negative variation (CNV) and reaction time. *Physiology and Behavior*, *4*, 351-357.
- Leuthold, H. & Jentzsch, I. (2001). Neural correlates of advance movement preparation: A dipole source analysis approach. *Cognitive Brain Research*, *12*, 207-224.
- Macar, F., Vidal, F., & Casini, L. (1999). The supplementary motor area in motor and sensory timing: evidence from slow brain potential changes. *Experimental Brain Research*, *125*, 271-280.
- Van Maanen, L., Brown, S., Eichele, T., Wagenmakers, E.-J., Ho, T., & Forstmann, B. U. (2011). Neural correlates of trial-to-trial fluctuations in response caution. *The Journal of Neuroscience*, *31*(48), 17488-17495.
- Van Rijn, H., Kononowicz, T. W., Meck, W. H., Ng, K. K., & Penney, T. B. (2011). Slow potentials in time estimation: The role of temporal accumulation and habituation. *Frontiers in Integrative Neuroscience*, *5*(91). doi: 10.3389/fnint.2011.00091

A Cognitive Model of Drivers Attention

Kerstin Sophie Haring (ksharing@fennel.rcast.u-tokyo.ac.jp)

Research Center for Advanced Science and Technology, The University of Tokyo
4-6-1, Komaba, Meguro-ku, Tokyo, 153-8904, Japan

Marco Ragni (ragni@cognition.uni-freiburg.de)

Lars Konieczny (lars@cognition.uni-freiburg.de)

Center for Cognitive Science, University of Freiburg
Friedrichstr. 50, 79098 Freiburg, Germany

Abstract

Cognitive architectures can account for highly complex tasks. One of the greatest challenges is understanding and modeling human driving behavior. This paper describes an integrated cognitive model of human attention during the performance of car driving. In this task, the attention process can be divided into at least three basic components: the control process, the monitoring process, and finally, the decision making process. Of these basic tasks, the first has the highest priority. All three phases are implemented in a cognitive model in the cognitive Architecture ACT-R 6.0. The model is able to keep a traffic lane, overtake another vehicle by lane change, identifies traffic signs and different situations emerging at crossroads.

Keywords: Driver behavior model; cognitive architecture; ACT-R; Attention

Introduction

Even for long-time practitioners driving a car is a highly complex task. This becomes evident by the still high number of accidents. E.g., in 2010 in Germany nearly 375.000 persons were injured in approximately 290.000 automobile accidents (Statistisches Bundesamt, 2011). In about 84% of all cases the cause of an accident could be traced back to driver errors (cp. Fig. 1). Nowadays passive safety systems like the airbag are reaching their technological limits and the focus shifts more towards active safety systems. Active systems, however, require exact knowledge about the driver, the vehicle, and the environment. To increase the acceptance of active intervention through the safety systems in cars, these systems should act in accordance to the driver. The driver and the human driving behavior must be considered for the future development of safety systems. Consequently, one focus of research is to analyze human behavior and predict possible errors.

We present the implementation of a cognitive driver model, simulating human attention and driving behavior. A driver model can be a powerful instrument with several possible fields of application, such as the development of intelligent driver assistant systems. The model is an adaption of Salvucci's (2006) driver model developed in the Cognitive Architecture ACT-R 5. Our model is implemented the newer version ACT-R 6 (Anderson, 2007) and using the standard ACT-R development environment running on an

open source LISP, which not only guarantees support and accountability, but also enables the research community to use the developed model for further research. It is able to keep a traffic lane, initiate and decide about a change of the lane in case of upfront traffic, identify prevalent situations at crossroads and react to traffic signs.

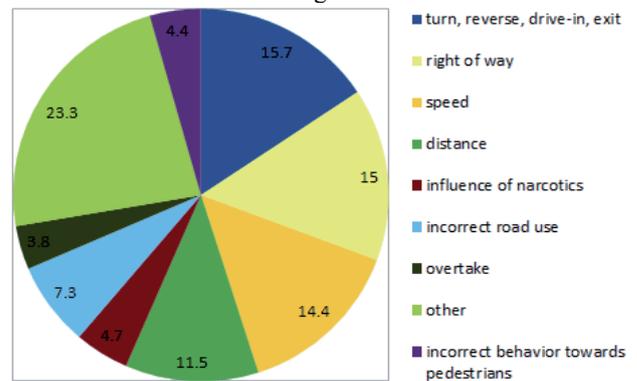


Fig. 1: Driver errors in automobile accidents with person injury (Statistisches Bundesamt, 2011).

Previous work

Most developed approaches can be distinguished into two classes: task specific and generic approaches. *Task specific approaches* such as Cosmodrive (Bellet et al., 2007) and Pelops (Benmimoun, 2004) reproduce the cognitive functions of a car driver. In contrast to task specific approaches, *generic approaches* can model various aspects of human behavior. Therefore, it is necessary for these architectures to include a theory of human information processing. Examples for such architectures in which driver models have been implemented are ACT-R (Anderson, 1993; Salvucci, 2006), SOAR (Aasman, 1995) and QN-MHP (Liu et al., 2006).

Previous models can be divided into three categories: First, early models concentrated mainly on steering and lane keeping. These models focus on the control process and are able to detect some cognitive aspects, but according to Boer (1999) they are highly dependent on difficult perceivable inputs from the environments. Second category comprises perception-action models which are through the perceptual constraints oriented closer on human behavior (e.g. Rushton

et al., 1998; Salvucci & Gray, 2004; Wilkie & Wann, 2003). Yet, these models do not allow for movement dynamics.

Finally, the third category includes models that are trying to unify the various aspects of a driving task and are therefore the most closely associated to the here presented work. These models not only explore and unify the various aspects of driving behavior, they also explore the generality of the cognitive architectures used for their development. Driver models were described by Aasman (1995) in the cognitive architecture SOAR and by Liu (1996) in Queuing Network-Model Human Processor (QN-MHP). Although these models already exist in other cognitive architectures and the central ideas remain the same in any architecture, the ACT-R model of a driver shows a broader spectrum of application (Salvucci 2001; 2006).

Salvucci (2006) developed a first integrated cognitive model of human driving behavior in ACT-R. He showed in his work the generality and the applicability using the cognitive architecture ACT-R for the specific task of driving. His model is designed to keep a standard vehicle on a multi-lane highway with moderate traffic. The model is also able to recognize the distance to a vehicle ahead and to make the decision for overtaking. As driving is a highly complex task and not readily implementable, this model has some limitations. The model solely was meant to interact with a highway environment without recognition of traffic signs, crossings or slip roads. An implementation limitation was the use of the previous version ACT-R 5.0 and its incompatibility to newer versions. It was also not possible to make the ACT-R model interact directly with a driving simulator.

The cognitive architecture

A cognitive architecture comprises theories about the operation mode of human information processing and aims at using procedures similar to humans. In other words, it describes a comprehensive computer model of human cognition. ACT-R (Anderson, 1993; Anderson 2007) is such a comprehensive theory of human cognitive capacities. It is also a modeling environment, used to describe human cognitive processes. Most of its basic assumptions are inspired by the progress of cognitive neuroscience. ACT-R is a framework in which the researcher can create models (programs) for different tasks. Running this model produces a simulation of human behavior. The main assumption of ACT-R is the representation of knowledge as either declarative or procedural knowledge. Declarative knowledge, consisting of facts, is represented in form of *chunks*, or small logical units which encode simple facts (e.g. the fact: “Berlin is in Germany”). Procedural knowledge, representing knowledge about how we do things, is represented in form of *production rules*, condition-action rules that generate a specific action (e.g. manipulate declarative knowledge) if the conditions of this rule are fulfilled.

In other words, ACT-R’s knowledge representation is split in two kind of memory modules. Modules can be

accessed through their buffers. The state of ACT-R at a given time is the content of the buffers at that time. Buffers are connected to the modules and are changed by production rules. Every buffer and (nearly) every module can be allocated to a cortex region. This enables an interesting mapping between buffers and neural processes (Anderson 2007).

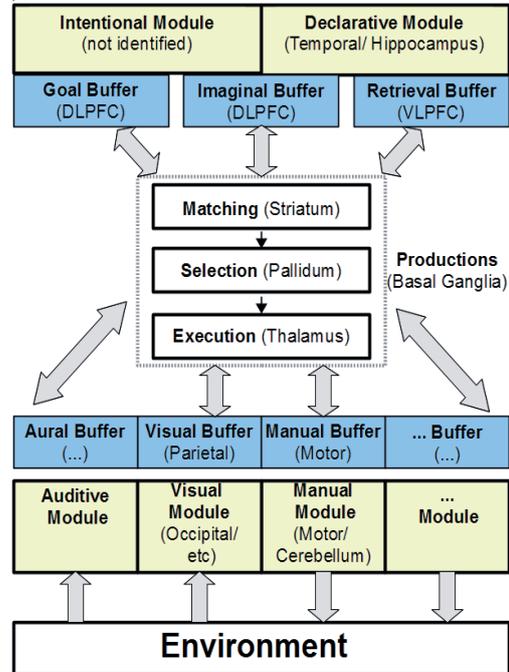


Fig. 2: The organization of information the cognitive architecture ACT-R (Anderson, 1993). The buffers contain information and are connected to modules associated with brain regions.

Cognitive model

We introduce now a computational model of human attention in a car driving task implemented in the ACT-R architecture. It models human attention and behavior for driving a car on a straight road, overtaking another vehicle by lane-change, identifying a traffic sign and crossroads.

Driver Modeling

The goal of this research was to develop an integrated driver model in the context of embodied cognition, task and artifact (ETA) framework. Byrne (2001) describes the ETA framework as understanding of interactive behavior based on the Cognition-Task-Artifact triad introduced by Gray (Gray & Altman, 2001). Interactive behavior is a function of the performed Task, the Artifact (instrument) by which the task is performed, and the Embodied Cognition, the cognitive, perceptual and motor capabilities by which a person acts through the artifact.

Cognitive modeling of human driving behavior should address all three components. An integrated model considers the driving related tasks (Task), the interface between the human and the vehicle (Artifact) and the

processes that execute the driving task on the vehicle (Embodied Cognition). The system must be specified regarding a detailed description of the artifact being used and the task to perform. Some successfully implemented and applied models only emphasize one or two of these components like the perception-action models of control of Fajen (Fajen & Warren, 2003), which provides a compact description of the behavioral dynamics of steering and obstacle avoidance, control-theoretic models like Donges (1978), dividing the steering task into a guidance and a stabilization level or machine-learning models, supporting automobile drivers steering by sampling an image, assessing the road curvature, and determining the lateral offset of the vehicle (Pomerleau & Jochem, 1996).

Driving is a continuously changing task of basic subtasks. These must be integrated and interleaved. This model uses three basic components, control, monitoring, and decision making (see Fig. 3), derived from the hierarchical control structure of Michon (1985). Michon identified three levels of skills and control for the driving task: operational (control), tactical (maneuvering), and strategic al (planning). He claims that a comprehensive model should take into account the various levels and also provide an information flow control that allows to switch from one level to the other.

The independent subtasks of a simple driving task (see Fig. 3) were implemented as *control*, the operational process controlling the input, *monitoring*, the tactical process interacting with the environment, and *decision making*, also analogous to the tactical level of Michon (1985), managing maneuvers like overtaking. These subtasks are processed serially. Every production of the top level goal *drive* has sub-goals, which incorporate the three components.

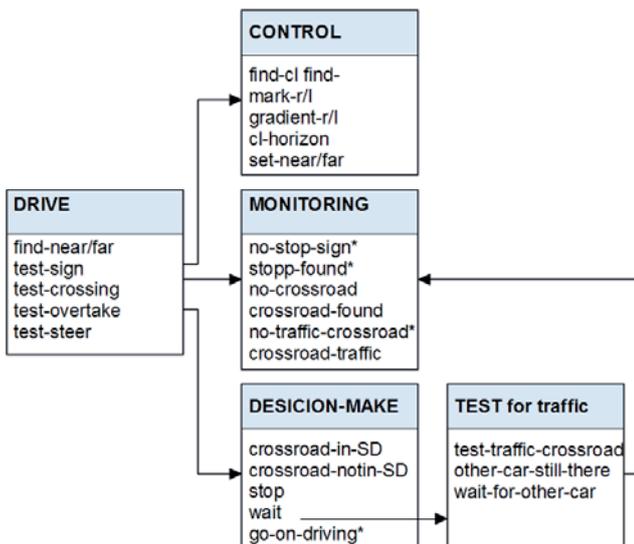


Fig. 3: Schematic representation of the production rules of the driver model in a simple crossroad scenario. The title of a box indicates the current goal and the corresponding production rules. The arrows show the flow of control and the asteriks the return to the parent-goal.

Development Environment The theory of ACT-R is embedded in the ACT-R software in form of Common Lisp functions. This model is implemented in Clozure CommonLisp 1.3 and the current version of ACT-R 6.0 under the operating system Ubuntu 9.04. In order to make the simulation environment interact with the ACT-R system, it was directly implemented in LISP with simple graphics and the extension with the LTK Lisp Toolkit. As it was not possible to make ACT-R directly interact with a driving simulator, we decided to use a Lisp-implementation of a driving environment.

Model Specification

As mentioned, the cognitive model of human attention integrated the three components control, monitoring and decision making. They are implemented as a loop of cognitive operations in the ACT-R serial processor.

The UML-Diagram in Fig. 4 shows the behavior of the cognitive model. To execute the task drive, the model runs through several states.

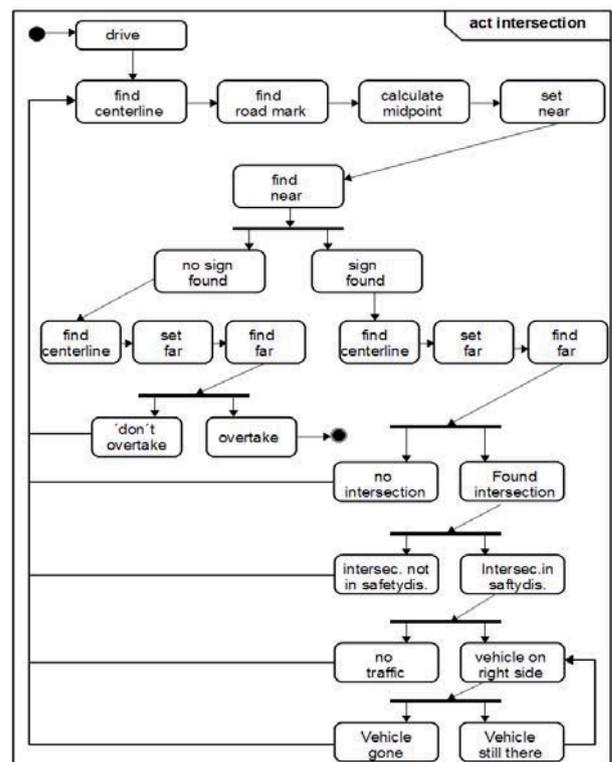


Fig. 4: UML-Diagram of the driver model

From the initial state, the model finds the road marks and sets the near point for stable navigation on the road. The model then fires a production rule screening for a traffic sign, changes the state according to the result and sets the far point. In our model, the near and far point are used as control components and explained in detail in the next paragraph. If the model reaches the state *find far* it can reach

the state *overtake* or will repeat the control loop. If there is special state like an intersection, the model tests for other given constraints and according to the result of this test, it will either go to another special state or repeat the control loop updating the near and far points.

A crucial advantage of the ACT-R architecture is that the three components control, monitoring and decision-making can be implemented directly. This takes into account human constraints and results in a cognitive adequate model of human attention.

Control

The control component of attention while performing a driving task manages the perception of lower level visual cues and the control over the vehicle (e.g., stopping). The model uses the simple concept of two salient visual attributes. This concept is based on earlier findings on locomotion (Llewellyn, 1971) and steering. Further research (Donges, 1987; Land & Horwood, 1995) describes steering as divided in two levels, guidance and stabilization, by using a „far“ and a „near“ region. Models of steering developed under this assumption have been proven to be consistent with empirical evidence.

The perception of this model is based on the perception of two salient visual points (Salvucci & Gray, 2004), a near and a far point. These two points are used for guidance, stabilization and also, to observe other salient attributes. For the here created artificial road environment, these two points account to recognize relevant aspects in any situation which may arouse during driving a car.

The near point determines the position on the road, which is in the middle of the center line and the border line. To identify the direction of driving, the far point is used and usually set on the vanishing point on the horizon or on the lead vehicle. The far point is also used to identify other situations and can be set on non-control points like traffic signs or approaching cars. Fig. 5 illustrates the near and far points.



Fig. 5: Near and far points for a straight road with a vanishing point and a road segment with a lead car.

The ACT-R architecture limits the employment of the control component by using a serial cognitive processor. The serial processing of the subtasks is typical for the human bottleneck of information processing. The resulting model is not an optimal model in a mathematical sense, but approximates human behavior.

In a driving environment, the majority of lower level visual control is keeping the vehicle in the middle of the road lane, for which the near point is used. Although the far

point is used to identify traffic signs, it mainly indicates the driving direction.

If the far point is not set on the vanishing point on the horizon, the model uses the combination of near and far point for determining the current scenario (see also Fig. 8 for an overview of implemented scenarios). If there is a lead vehicle, the distance between the two points is determined, and in case it falls below a certain safety distance, the model can react according to that (e.g. through slowing down or overtaking). In a crossroad scenario without an approaching car from the right hand side, the model will set the far point on the vanishing point of the horizon and continue driving. After that, this model will not look again for another car at the crossroad, which is surely an issue for future implementations. In case there is a vehicle or a stop sign, the stopping of the car is implemented here by setting the far point onto the near point. The model will continue a loop until the other vehicle is not on the crossroad anymore and out of the safety distance.

Monitoring

After the control component, the monitoring is one of the most important. Here, the environment is continuously captured (e.g. the model looks for a traffic signs) and updated in the declarative memory. In the here implemented driving environment, the situation awareness mainly focuses on other vehicles around, change of the scenario (from straight road to crossroad), or traffic signs. The model shifts the focus of visual attention towards a certain object which is then encoded as visual attribute. The shift could be based on a random-sampling model, checking the different environment areas with a probability p , which has been successfully done by Salvucci (2006). Here, the model monitors particular directions and visual attributes (e.g. other vehicles, center line) by an attention shift. The encoded attribute is noted in the declarative memory. As ACT-R has a build-in memory decay mechanism, it might be possible to predict driver errors because the chunks encoding the current environment decay and can be forgotten if not updated continuously. Another source of possible driver errors could be the potential failure in encoding relevant information (e.g. to overlook a traffic sign or a vehicle).

Decision Making

The information provided by the control and monitoring component is used to determine if and what decisions must be made on the tactical level concerning the maneuvering (e.g. stopping or overtaking). The most common decision making might be whether to stop or to continue driving. This decision depends on the traffic sign or on other vehicles. As described earlier, the execution of stopping corresponds simply to the use of the near and far points encoding current position and relevant aspects of the environment. In order for the model to produce a decision making process similar to humans, encoding a visual attribute and shifting visual attention cannot occur at the

same time. For this model, the focus of attention is for example either on the near or far point or encoding a traffic sign. This restriction through serial processing seems to be a drawback in the sense of mathematical optimal behavior, but it describes the bottleneck typical for the human information processing (Anderson et al, 2004). Through the implementation of this restriction, it is possible to mimic human cognitive capacities, simulate the dynamic nature of human driving behavior, and therefore a cognitive adequate model of human driving behavior is produced.

The knowledge representation comprehends declarative knowledge in chunks and procedural knowledge in production rules. For example, the scenario at a crossroad was implemented in 73 explicit production rules, which are highly detailed and is therefore open to future extensions of the model. The control of attention in the ACT-R architecture is achieved through three different methods of shifting attention: First by specific locations or directions, second by specific characteristics, and third by objects, that have not been in in the focus of attention yet.

The combination of these methods of attention shift enables the model to create complex search strategies through the production rules.

Results and Discussion

We present a simulation environment and a cognitive model of driver attention during car driving that is able to interact during run-time. In this work, two driver models were developed. The first model is able to reliably keep the traffic lane on a two-lane road and initiate a lane change followed by overtaking another vehicle. It identifies another vehicle and decides to overtake it if the safety distance falls below a certain distance (Fig. 6, scenario 1 and 2). The second model builds up on the first model and extends its functionality by identifying crossroad (Fig. 6, scenario 3), traffic signs and vehicle on the right hand side which have right of way (scenario 4, 5 and 6).

To obtain an integrated driver model of human driving behavior, it is essential to develop models in an architecture which is not task specific and can also model human behavior also in a different context, like ACT-R. This model is a first attempt to recognize, still simplified, traffic signs and crossroads. The development of an integrated driver model makes a first step towards the vision of accident-free driving. A majority (over 80%) of the automobile accidents are caused by the driver themselves. Fig. 1 shows the human errors while driving. Nearly 16% of the accidents happen while turning and during exit, followed by disregarding the right of way (15%) and not-adapted speed (15%). Theoretically, the cognitive driver model could give a deeper insight for around 30% of the human errors while driving. However, it has to be taken into account that the model is still interacting with a simplified environment and not yet taking into account driver's prior experience, which could be implemented by an increased attention in potentially high accident risk situations. Our driver model is one approach to integrate operational (lower-level) and

tactical (higher-level) aspects in the framework of the ACT-R architecture. The model and the environment do not present a complete picture of driver behavior yet, but they form a base to extend the ETA framework in any direction.

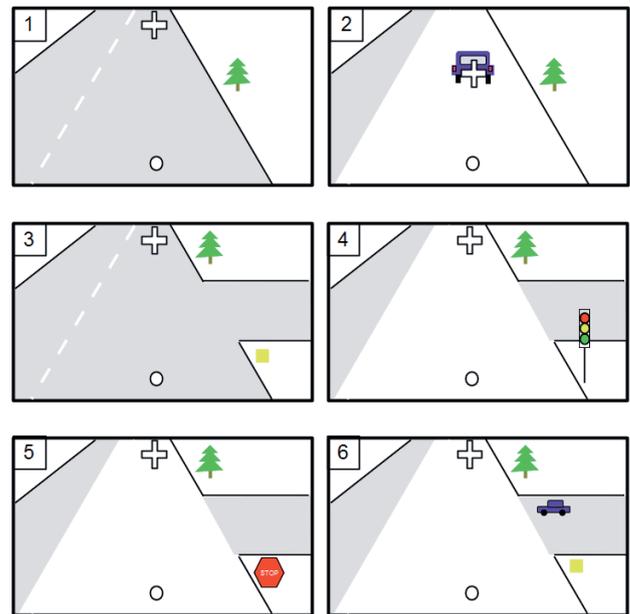


Fig. 6: Standard situations while driving a car which can be handled by the cognitive drive model.

The aspect of limited cognitive resources is one of the main factors for the adequacy of the model. Based on the implemented bottleneck, the three components control, monitoring, and decision making, have to share cognition. If the model is occupied with attention shift, it cannot simultaneously update the near point. Also, the model can only fire on production rule at a time and only one visual operation can be executed at a time. These processes take a certain time. For example, in the standard implementation in ACT-R, one firing of a production rule requires 50ms. This enables the researcher to compare the produced data with human data, because the ACT-R architecture produces an output file. This file contains the time, the active buffer and the according event. This study did not validate the model data so far. Future research could compare the output file data with human data, specially compare the attention shift of the model to human drivers over eye-tracking and the reaction times. However, for this validation, it must be possible from the technical side to either connect the ACT-R model directly to the simulation environment or to produce the same output file for the human data as the model does. Also, only most critical parts of key scenarios can be validated as no single method is sufficient enough to understand the complex task of human driving behavior yet.

Conclusion and Outlook

The progress to date in the development of cognitive architectures has been impressive, yet scientific gaps,

technical challenges and practical issues remain. On one hand, cognitive models help to develop an understanding of driver behavior and aim to provide a theoretical account for human attention while driving. On the other hand, they are powerful and practical tools when implementing human-centered design and real-world applications. First steps towards the examination of the source of human mistakes through distraction from the primary driving task through secondary tasks like dialing a phone have been taken (Salvucci, 2001) showing the feasibility of the architecture for these task and possible extensions.

The ACT-R architecture enables to elucidate interesting aspects and provides a theory of human attention while driving. At the same time, human attention during driving is a challenging task for the ACT-R cognitive architecture. It shows the still existing limitations beyond basic laboratory tasks and pushes the research community to expand the architecture towards more complex and finally real-world tasks.

Acknowledgments

We are grateful for the kind advice and assistance of Prof. Bernhard Nebel, the support of Prof. Wolfram Burgard (University of Freiburg), and for intensive discussions with Dario Salvucci (University of Philadelphia). This work has been partially supported by a grant from the DFG to MR (Project R8-CSPACE in the SFB/TR8 "Spatial Cognition").

References

- Aasman, J. (1995). Modeling driver behavior in Soar. In: Leidschendam, The Netherlands: KPN Research
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C. & Qin, Y. (2004). An Integrated Theory of the Mind. *Psychological Review*, 111, 1036
- Anderson, J. R. & Lebiere, C. (1998). The atomic components of thought. Mahwah, NJ: Lawrence Erlbaum.
- Bellet, T., Bailly, B., Mayenobe, P., & Georgeon, O. (2007). Modelling Driver Behavior in Automotive Environments. Critical Issues in Driver Interactions with Intelligent Transportation Systems. *Cognitive Modelling and Computational Simulation of Driver Mental Activities*.
- Benmimoun, A. (2004). Der Fahrer als Vorbild für Fahrerassistenzsysteme? Ein fahrermodellbasierter Ansatz zur Entwicklung von situationsadaptiven FAS. 13. Aachener Kolloquium Fahrzeug- und Motorentechnik
- Boer, E. R. (1996). Tangent point oriented curve negotiation. *IEEE Proceedings of the Intelligent Vehicles 96 Symposium*
- Boer, E. R. (1999). Car following from the driver's perspective. *Transportation Research – Part F*, 2, 201-206
- Byrne, M. D. (2001). ACT-R/PM and menu selection: Applying a cognitive architecture to HCI. *International Journal of Human-Computer Studies*, 55, 41-84
- Donges, E. (1987). A two-level model of driver steering behavior. *Human Factors*, 20, 691-707
- Fajen, B. R., & Warren, W. H. (2003). Behavioral dynamics of steering, obstacle avoidance, and route selection. *Journal of Experimental Psychology: Human Perception and Performance*, 29, 343-362
- Gray, W. D., & Altmann, E. M. (2001). Cognitive modeling and human-computer interaction. *International encyclopedia of ergonomics and human factors*, 1, 387-391, Taylor & Francis, Ltd.
- Land, M., & Horwood, J. (1995). Which part of the road guide steering? *Nature*, 3, 77, 339-340
- Liu, Y. (1996). Queuing network modeling of elementary mental processes. *Psychological Review*, 103, 116-136
- Liu, Y., Feyen, R., & Tsimhoni, O. (2006). Queuing Network-Model Human Processor (QN-MHP): A computational Architecture for Multitasking Performance in Human-Machine Systems. *ACM Transactions on Computer-Human Interaction* 13, 37870
- Llewellyn, L. (1971). Visual guidance of locomotion. *Journal of Experimental Psychology*, 91, 245-254
- Michon, J. A. (1985). A critical view of driver behavior models: What do we know, what should we do? *Human behavior and traffic safety*, 485-52, Plenum Press
- Pomerlau, D., & Jochem T. (1996). Rapidly adapting machine vision for automated vehicle steering. *IEEE Expert*, 112, 19-27
- Reid, L. D., Solowka, E. N., & Billing, A. M. (1981). A systematic study of driver behavior steering control models. *Ergonomics*, 24, 447-462
- Rushton, S. K., Harris, J. M., Lloyd, M.R., & Wann J.P. (1998). Guidance of locomotion on foot uses perceived target location rather than optic flow. *Current Biology*, 8, 1191-1194
- Salvucci, D. D. (2001). Predicting the Effects of In-Car Interface Use on Driver Performance: An Integrated Model Approach. *International Journal of Human-Computer Studies*, 55, 85-107
- Salvucci, D. D. (2006). Modeling Driver Behavior in a Cognitive Architecture. *Human Factors*, 48, 362-380
- Salvucci, D. D., Liu, A., & Boer, E. R. (2001). Control and monitoring during lane changes. *Vision in Vehicles*, 9
- Salvucci, D. D., & Gray, R. (2004). A Two-Point Visual Control Model of Steering. *Perception*, 33, 1233
- Statistisches Bundesamt (online 20.12.2011) www.destatis.de
- Wilkie, R. M., & Wann, J. P. (2003). Controlling steering and judging heading: retinal flow, visual direction and extra-retinal information. *Journal of Experimental Psychology: Human Perception and Performance*, 29, 363-378

Cognitive modeling of different processing modes in task switching: toward an explanation of the effect of aging on switching cost

Stéphane Deline (stephane.deline@gmail.com)
LPE-CRPCC, Université Rennes 2.
Place du Recteur Henri Le Moal, 35043 Rennes, France

Jacques Juhel (jacques.juhel@univ-rennes2.fr)
LPE-CRPCC, Université Rennes 2.
Place du Recteur Henri Le Moal, 35043 Rennes, France

Keywords: Cognitive aging; task-switching; cognitive control.

Introduction

Cognitive aging is associated with a decrease of executive control ability that results in impaired performance in inhibition tasks (Hasher & Zacks, 1988) or task-switching (Mayr, Spieler, & Kliegl, 2001). Regarding task-switching, mixing costs are generally greater for elderly than for young people (Wasylyshyn, Verhaeghen & Sliwinski, 2011). One explanation of this phenomenon is that individuals fail to maintain task representations in a sufficient active state, (Engle & Kane, 2004). However, this hypothesis can't explain the observation of an equivalent switching cost between young and elderly (Wasylyshyn et al., 2011). Braver and West (2008) made an additional assumption of the effect of aging which presumes a declining ability to maintain goals representations. More specifically, this hypothesis supposes a decrease of the efficacy of proactive control mechanisms (controlled orientation or preparation of activities), resulting in a greater tendency to initiate reactive control processes (on-line processing).

The aim of this study is to test with computational modeling to what extent the Braver and West (2008) hypothesis can account for age and individual differences in task-switching tasks.

Method

The task used in this study runs as follows: cue presentation ("+" or "-") for 1 sec. ; 750 ms later target presentation ("black" or "white" disk); manual response (pressing one of two buttons on a case-response) and disappearance of the target for 1 sec. ; Onset of the next cue. The experimental condition depends on the cue appeared. In condition A, called "congruent" (cue "+"), the participant must press the button which matches the target color (ie "white" to "white" "black" to "black"). In condition B, called "incongruent" (cue "-"), he must press the opposite color (ie "white" for "black" "black" to "white"). The experience includes a first familiarization phase (homogeneous block of 17 trials A; homogeneous block of 17 trials B; mixed block of 17 trials ABAB...) followed by the experimental phase (mixed block of 209 trials ABAB...).

Cognitive model

The cognitive functioning underlying task resolution is modeled using the ACT-R architecture (Anderson, 2007). First, the model incorporates visual, memory and decision processes (Altmann & Gray, 2008), as well as more specific processes of interference (Oberauer, 2002) and top-down cognitive control processes (Meiran, Kessler & Adi-Japha, 2008). Secondly, it incorporates two different modes of task processing, based on two main theories of task-switching discussed in the literature. The first one, called "on-line", is based on the compound-cue theory (Logan & Bundesen, 2003) which supposes that the combination of the stimulus and a simple representation of the cue is sufficient to select effectively the correct answer. The second, called "preparatory", is inspired by the task-switching reconfiguration theory (Rogers & Monsell, 1995) which assumes that individuals use more complex task representations to guide the selection of the response (Dreisbach & Haider, 2009). In this model, the use of each processing mode depends on the type of cue representation (simple or complex) extracts from declarative buffer.

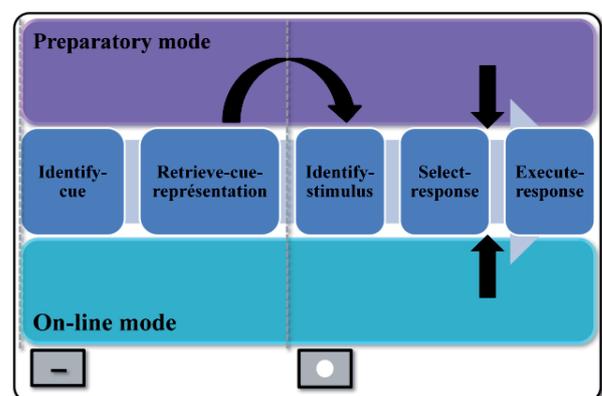


Figure 1: Main steps, control processes and task processing modes implemented in the cognitive model. (Black arrows indicate control processes)

Aging hypotheses testing

Several parameters can be manipulated to test cognitive aging hypotheses: 1) the latency factor *lf* which influences the time to extract knowledge, 2) the goal activation parameter *ga* that modifies the amount of activation spread to knowledge in declarative memory, 3) the noise parameter *ans* which introduces noise in the activation level of knowledge, or 4) the probability of execution of the two processing modes implemented. Different hypotheses are tested according to parameter(s) manipulated: slower processing speed (1), the reduced capacity of working memory (2), increased noise cognitive (3) or the initiation of control processes preferred reagent (4).

Results

The results presented in this work are discussed under the Braver and West (2008) assumption. The parameters of the model manipulated are the activation levels of cue representations that determine the probability of initiation of each processing mode implemented. It consists of a large decrease (resp. increase) of the probability of initiation of the preparatory mode, which increases (resp. decrease) the switching cost simulated (latency difference between incongruent and congruent trials, in mixed condition). This effect is further accentuated if the parameter value *lf* is high (ie slowing). The analysis of convergence between simulated and empirical data obtained from a sample of 13 women and 10 men aged 20 to 83 years ($M = 46.9$ years, $SD = 20.2$; $MMS > 26$ for people over 65 years old) indicates that the Braver and West (2008) hypothesis for a decrease with aging of the probability of initiation of proactive control processes -associated with slowing- can account for the increase of sensitivity to constraint changes observed empirically in older individuals.

References

- Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* New York: Oxford University Press, USA.
- Altmann, E. M., & Gray, W. D. (2008). An integrated model of cognitive control in task switching. *Psychological Review*, 115(3), 602-639.
- Braver, T. S., & West, R. (2008). Working memory, executive control, and aging. *The handbook of aging and cognition* (3rd ed.), 311-372.
- Dreisbach, G., & Haider, H. (2009). How task representations guide attention: Further evidence for the shielding function of task sets. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 35(2), 477-486.
- Engle, R. W., & Kane, M. J. (2004). Executive attention, working memory capacity, and a two-factor theory of cognitive control. *The Psychology of Learning and Motivation*, 44, 145-199. In Ross (Ed.). NY: Elsevier.
- Hasher, L., & Zacks, R. T. (1988). Working memory, comprehension, and aging: A review and a new view. *The Psychology of Learning and Motivation*, 22, 193-225. In G.H. Bower (Ed.), New York, NY: Academic Press.
- Logan, G. D., and Bundesen, C. (2003). Clever homunculus: Is There a year endogenous act of control in the explicit task-cuing procedure? *Journal of Experimental Psychology: Human Perception and Performance*, 29 (3), 575-599.
- Oberauer, K. (2002). Access to information in working memory: Exploring the focus of attention. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 28, 411-421.
- Mayr, U., Spieler, D., & Kliegl, R. (2001). Ageing and executive control: Introduction to the special issue. *European Journal of Cognitive Psychology*, 13, 1-4.
- Meiran, N., Kessler, Y., & Adi-Japha, E. (2008). Control by action representation and input selection (CARIS): a Theoretical Framework for task switching. *Psychological Research*, 72 (5), 473-500.
- Rogers, R. D., & Monsell, S. (1995). Costs of a predictable switch Between simple cognitive tasks. *Journal of Experimental Psychology: General*, 124 (2), 207-231.
- Wasylyshyn, C., Verhaeghen, P., & Sliwinski, MJ (2011). Aging and task switching: A meta-analysis. *Psychology and Aging*. 26 (1), 15-20.

A LIDA-based Model of the Attentional Blink

Tamas Madl (tamas.madl@gmail.com)

Department of Philosophy (Cognitive Science), University of Vienna
Dr. Karl Lueger Ring 1, Vienna, 1010 Austria

Stan Franklin (stan.franklin@memphis.edu)

Institute for Intelligent Systems, University of Memphis
Memphis, TN 38152 USA

Abstract

The attentional blink (AB) refers to the impairment in consciously perceiving the second of two targets presented in close temporal proximity (200 – 500ms) in a rapid serial visual presentation paradigm. The present paper is a preliminary report describing a conceptual and partially computational model of the AB based on the LIDA (Learning Intelligent Distribution Agent) cognitive architecture. The model aims to provide a biologically plausible explanation of the AB, explaining a wide range of AB-related phenomena, among other mental phenomena accounted for by LIDA. Computational results in a basic visual AB paradigm are presented and compared to human data.

The Attentional Blink

When subjects are asked to identify two targets separated by a short time (200-500ms) in a stream of distractors, an Attentional Blink (AB) occurs - subjects often fail to report the second target (see Fig 1). In this paper we will focus on the AB in rapid serial visual presentations (RSVP) of pictures (however, the AB has been shown to occur across a wide range of stimuli types and modalities – see (Martens & Wyble 2010)).

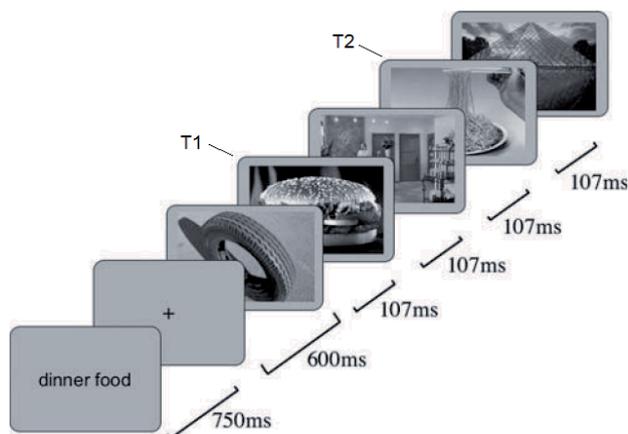


Fig 1. The visual attentional blink paradigm (from Potter et al. 2010). Subjects who have to identify targets in a stream of images often fail to report the second target (T2) if it is presented shortly after the first target (T1).

Brain related evidence has shown that during an AB task, both targets are processed at least perceptually, regardless of conscious reportability - at least the first 150ms of neural activity exhibits a normal pattern (Martens & Wyble 2010). An fMRI study conducted by (Marois et

al. 2004) showed parahippocampal place area activations (associated with high-level scene representations) even in non-conscious T2 targets. However, EEG studies have revealed the electrophysiological activity that correlates with the AB – the N2pc ERP¹ component, occurring about 200ms poststimulus and associated with the allocation of attention to targets - is suppressed at short temporal distances between T2 and T1. Also, in trials where T2 cannot be perceived because it is presented shortly after T1, the P3 component - associated with working memory consolidation - is not elicited (Martens & Wyble 2010), (Dux & Marois 2009). The above evidence implies that the AB has to occur at a later stage of processing (later than perceptual recognition, and after 150ms).

Apart from this finding, a number of attentional blink related phenomena have been found, some of which have proven hard to explain – no complete, formal account for all of these has been found yet (Dux & Marois 2009). Elaborating on all the AB-related effects that have been identified would exceed the scope of this paper (See Martens & Wyble 2010 and Dux & Marois 2009 for more phenomena). The following phenomena have been chosen to highlight the ABs main properties, and to show that while only a simulation of the basic AB paradigm is presented in this preliminary paper, our LIDA-based model can provide much wider explanations. Future work will be required to computationally simulate and verify them.

1) Lag-1 sparing. Paradoxically, T2 can be reported with high accuracy if presented shortly after T1 (about 100ms after T1; “lag n” describes the temporal distance between the targets) (Martens & Wyble 2010).

2) Spread lag-1 sparing. Multiple targets can be reported as long as they are presented in immediate succession – it has been observed that target reports were accurate even for four successive targets (Olivers et al. 2007).

3) Posttarget intrusion. Varying the experimental conditions revealed that the AB only occurs if T2 is backward masked (Giesbrecht & Di Lollo 1998). Often, this mask or distractor succeeding T2 can be reported even if T2 cannot, implying that the distractor somehow interferes with the reporting of the target (Chun 1997).

4) Whole report attenuates the AB. The accuracy of reporting stimuli is high when subjects are asked to report all stimuli (whole report). However, a significant accuracy

¹ Event-Related Potential, brain activity directly resulting from and time locked to a stimulus

drop at lags 2 – 4 (an AB) can be observed for the same stimuli sequence if subjects are required to report only two targets in the sequence (Potter et al. 2008).

5) Increasing T2 salience/arousal attenuates the AB. If the salience of the second target is increased, it can be reported more accurately, although an AB effect can still be observed (Martens & Wyble 2010). Emotional arousal (but not valence) also alleviates the AB (Anderson 2005).

6) Task-irrelevant cognitive load attenuates the AB. If the stimuli are presented together with a background field of moving or flickering dots, much smaller drops in accuracy are observed at AB-relevant lags (Arend et al. 2006). The AB is also attenuated if subjects are asked to listen to task-irrelevant music or think about their holiday (Olivers & Nieuwenhuis 2005).

7) Target Confusion. The order targets were presented in is often confused for temporally adjacent targets (i.e. during lag-1 or spread lag-1 sparing) (Dux & Marois 2009), (Chun 1997).

8) AB without T1 masking. Although T2 masking is necessary to obtain an attentional blink, recent studies found that there is an (attenuated) blink even if T1 is unmasked (i.e. if there is no distractor between T1 and T2) (Nieuwenstein et al. 2009).

As will be described below, the LIDA model's attention mechanism is capable of explaining these phenomena, and is detailed enough to make a computational implementation reproducing an actual AB experiment feasible.

Attention in the Brain

The following brain areas play a role in top-down attentional control. The visually selective regions of the posterior parietal cortex – the intraparietal sulcus in humans and lateral intraparietal area in primates -, which contain coarse representations of spatial topography, and are also involved in controlling eye movements (saccades) and directing them towards targets. In this area, neuronal activity correlated with the voluntary allocation of attention can be observed, and leads to greater target stimulus selectivity based on spatial location and/or salience (Knudsen 2007), (Serences & Yantis 2006). The frontal eye fields (FEF) in the pre-frontal cortex are involved in saccade control as well, but have also been shown to play a role in representing the current locus of attention (Serences & Yantis 2006) – FEF neurons can be covertly selective to targets, without shifting gaze (Thompson et al. 2005). Also, when a particular stimulus is attended to and conscious, neurons representing the target in sensory areas, in the PPC and in the PFC exhibit synchronized discharges in the gamma band (Knudsen 2007), (Doesburg et al. 2009).

The superior colliculus (SC), like FEF, mediates both overt saccades and covert shifts of attention (Serences & Yantis 2006). Corollary discharges associated with eye saccades occur in the SC and propagate to the FEF, via the mediodorsal thalamic nucleus, and shift the locations of visual receptive fields in FEF before each saccade (Knudsen 2007).

Recently, the locus coeruleus (LC) – norepinephrine (NE) – system has been shown to influence top-down attentional selection (Nieuwenhuis et al. 2005)(Sara 2009), (Warren et al. 2009). LC neurons exhibit increased activity during the processing of motivationally salient targets, leading to the release of NE in widespread cortical projection areas, among others in the forebrain. This increased NE presence increases the responsibility of target neurons, enhancing signal detection and stabilizing a neural representation against noise or interference. Thus, LC activity can facilitate the processing of a target stimulus (Nieuwenhuis et al. 2005), (Warren et al. 2009), (Aston-Jones & Cohen 2005).

Shortly after target processing, there is an increase of LC neuronal activity (Fig 2A). The LC is autoinhibitory – increased activity during target processing is followed by a

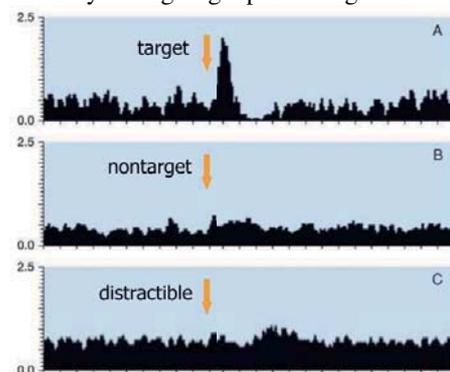


Fig 2. Peri-event time histograms (PETHs) for a typical individual monkey LC neuron in response to various events during performance of the signal detection task, accumulated over 100 sweeps of activity. Note the increased activity during target processing (panel A). Adapted from (Aston-Jones & Cohen 2005).

period of reduced activity and thus reduced NE release. Non-target stimuli do not elicit significant changes in LC activity (Fig 2B). Interestingly, tonic (regular spiking) LC activity levels were significantly higher when the animal was less focused on the task (Fig 2C), but there was also a much smaller target-locked phasic response in this case. Behaviorally, these elevated LC activity levels have led to more frequent false alarm errors (Aston-Jones et al. 1999).

These properties of the LC-NE system play a key role in LIDA's attentional blink model.

A LIDA-based Attentional Blink Model

The LIDA cognitive architecture is based on prevalent neuroscience and cognitive science theories (Global Workspace Theory, situated cognition, perceptual symbol systems, working memory, memory by affordances, transient episodic memory, etc. - (Baars & Franklin 2009)). It has been implemented computationally, and has been shown to perform simple psychological tasks with mechanisms similar to humans (Madl et al. 2011). LIDA's cognitive cycle has the purpose of selecting an appropriate action based on the perceived current situation, and has the following components (Madl et al. 2011):

- 1) *Perception*. Sensory stimuli are received and stored in a sensory buffer in the Sensory Memory. Percepts, emotions, and concepts are represented by nodes in the Perceptual Associative Memory (PAM). These are based on perceptual symbols (Barsalou 1999); their activations reflect recognition confidence as well as bottom-up salience.
- 2) *Percept to preconscious buffer*. Recognized percepts are stored in the preconscious buffers of LIDA's long-term working memory (Workspace).
- 3) *Local associations*. Local associations are automatically retrieved from the Transient Episodic and Declarative Memory using the Workspace contents.
- 4) *Competition for consciousness*. Attention codelets² (AC) in the Attention Codelet Module (ACM) view long-term working memory, and compete to bring novel, relevant, important, urgent, or insistent events to consciousness.
- 5) *Conscious broadcast*. A coalition of codelets, typically an AC and its content of PAM nodes, gains access to the Global Workspace (GW) and has its content broadcast consciously.
- 6) *Recruitment of resources*. Relevant behavioral schemes in Procedural Memory respond to the conscious broadcast.
- 7) *Activation of schemes in the Procedural Memory*. Schemes are instantiated in the Action Selection module, and receive activation, based on the conscious contents.
- 8) *Action chosen*. The Action Selection module chooses a single scheme from the newly instantiated schemes and remaining previously active schemes.
- 9) *Action taken*. The execution of the action of a scheme results in external or internal consequences, or both.

The major components implementing top-down attention in the LIDA model are the GW module and the ACM. Feature detectors (corresponding to feature-sensitive neurons in the visual cortices) pass activation to their corresponding PAM nodes, which represent objects (or categories, concepts, ...) and could correspond to neuronal ensembles in the inferior temporal cortex, which contain object category information (Liu et al. 2009). The resulting activation of PAM nodes will depend on the number of relevant features, as well as the salience of those features.

The ACM contains ACs, which create coalitions from important or relevant percepts in the Workspace. The coalition with the highest activation will be broadcast consciously. Coalition activation depends on four factors: a) the activations of the percepts it contains, b) the base level activation of the AC, c) the modulatory activation of the ACM and d) a matching factor on how well the percept matches the pattern that the Codelet is looking out for. The computational implementation of the LIDA AB Agent also contains a fifth factor, e) stochastic noise, which is added to account for extraneous, uncorrelated afferent activity (Knudsen 2007), (Nieuwenhuis et al. 2005).

² The term codelet refers generally to any small, special purpose processor or running piece of computer code. The concept is essentially the same as Baars' (1988) processors or Minsky's (1988) agents. The term was borrowed from (Hofstadter & Mitchell 1994).

The first factor a) corresponds to bottom-up salience in the brain, as described above. The second, b), the base level activation, depends on how useful the AC has been in the past and facilitates attentional learning.

The third factor, c), is the modulatory activation of the ACM. It has been proposed many times in attention literature that human attentional processing is limited for targets presented in short succession - observable, among others, in an AB paradigm -, presumably because of a suppression of attentional enhancement of subsequent stimuli during the processing of a target (Nieuwenhuis et al. 2005), (Wyble et al. 2009), (Olivers & Meeter 2008). The modulatory activation reflects this mechanism, and regulates attentional enhancement of stimuli by increasing or decreasing the activation of coalitions in the Global Workspace. The most probable neural counterpart of this regulatory activity is the LC, which can enhance target processing through the release of NE in the forebrain (LC activity was proposed to play a role in the attentional blink by (Nieuwenhuis et al. 2005)). Similarly to LC neuron activity, the ACM activation at first increases upon processing a relevant or important target, followed by a period of low activation which is similar to the posttarget refractory-like autoinhibition exhibited by the LC (Fig 3 bottom). The ACM activation is governed by a function derived from interpolating LC PETH data (Aston-Jones & Cohen 2005).

The fourth parameter d) influencing coalition activation is a matching factor that is based on how well the percept in a coalition matches the pattern sought by the AC that creates the coalition. This accounts for the finding that in some cases, nontargets are attended to and reported instead of the targets if they are similar or share a common salient feature (Martens & Wyble 2010), (Lavie & Cox 1997), (Bichot & Schall 1999) although with less probability and less selective neuronal activation (Duncan et al. 1997).

LIDA's attentional mechanism can provide a computational explanation for the attentional blink and related findings. Two major reasons are proposed to account for the performance drop at intervals of 200ms – 500ms between the two targets (see Fig 3A bottom): a) the posttarget refractory-like period of the ACM activation, which leads to reduced target activations after ~200ms, and b) the discrete, competitive conscious broadcast mechanism (Baars & Franklin 2009).

For the current description, an RSVP attentional blink paradigm with images is assumed (see Fig 1). Stimuli are presented to the LIDA agent at a rate of one image every 107ms. The agent's task is to report target images pertaining to a specific target (in this case, vehicles), which means that there are at least two ACs, looking out for targets (vehicles) and distractors, respectively. This is also the paradigm used for the implementation of the LIDA Attentional Blink agent. If only a single target is presented, that target is added to a coalition by the Target Attention Codelet (TAC), will win the competition for consciousness since there is nothing that could compete with it, and can be consciously reported.

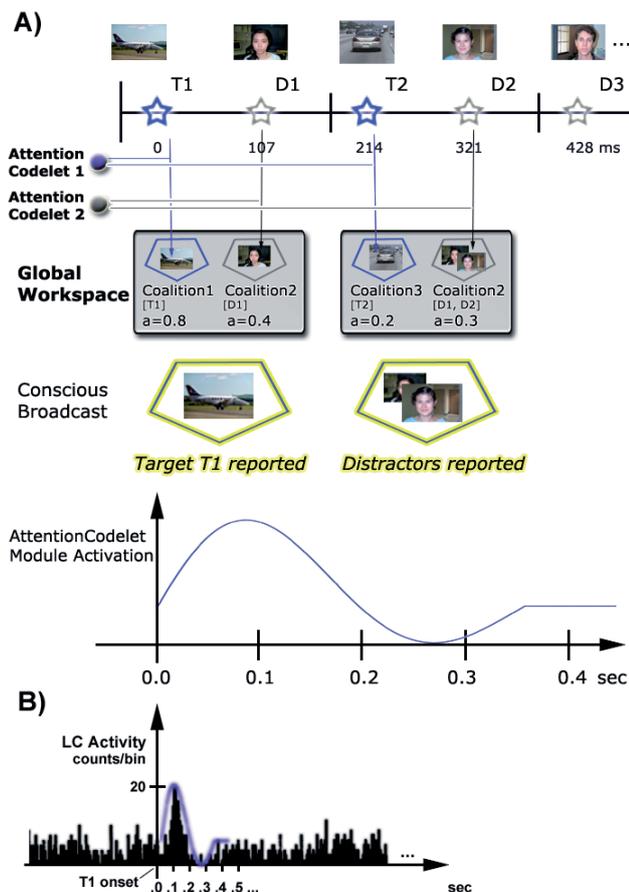


Fig 3. A) The attentional blink at lag 2. T_n and D_n refer to targets and distractors, respectively. The vertical black lines intersecting with the timeline on top represent the approximate borders of LIDA cognitive cycles. AC1 is looking out for targets, and AC2 for distractors, adding them to Coalitions in the Global Workspace. The coalitions have to compete for consciousness, and the one with the highest activation is broadcast consciously. The reason the agent fails to report T2 is that in the second cognitive cycle, Coalition 2 (containing the distractors) wins the competition for consciousness. B) LC activity - PETH of a monkey LC neuron during target processing.

If two targets are presented in an RSVP of images at lag-1, without a distractor, both targets are perceived in the first 200ms – before the refractory-like period of the ACM – and they are both added to a target coalition by an AC looking out for targets. This TAC has higher base level activation than the Distractor Attention Codelet (DAC). Thus the targets will win the conscious broadcast and can be reported consciously. Possible subsequent targets are also added to the target coalition by the same AC, which adjusts the coalition activation based on the factors described above and on the previous coalition activation – this accounts for the spread lag-1 sparing effect.

At lags 2 and 3, the second target sometimes cannot be reported consciously because a coalition containing

distractors wins the competition for consciousness instead of T2 (see Fig 3 and 4). The reason for the low activation of the target coalition is the low ACM activation at this point in time (due to the refractory-like period, see Fig 3 bottom). D2 is added to the distractor coalition by the DAC, and the coalition activation is updated. The distractor coalition is also modulated with a lower ACM activation, but will come out with a higher activation because a) depending on the timing of the presentation, the ACM activation might be higher at the point the distractor is perceived than at the point when the target is perceived, and b) since the distractor coalition was created upon perceiving D1, at which point the coalition activation was higher (0.4 in Fig 3, due to the high ACM activation at that point).

At lag 4, the ACM activation has regenerated to its initial level of activation, and T2 can be reported with a high level of accuracy again (the T2 accuracy at lag 4 approximately equals T2 accuracy at lag 1 in this paradigm, see Potter et al. 2010).

Results

LIDA's attentional mechanism conceptually accounts for all of the AB-related phenomena described above:

1-2) Lag-1 and spread lag-1 sparing. See above.

3) Posttarget intrusion. During the blink, the distractor succeeding T2 often can be consciously reported even if T2 itself cannot (see Fig 3).

4) Whole report attenuates the AB. In case of an instruction to report the entire RSVP sequence, a different Attentional Codelet would be required, which would move every presented image into the Global Workspace and into the same coalition – every image would be a target. Thus, for short RSVP sequences, every image could be reported and no AB could be observed (if the sequence is too long, activation decay could lead to “forgetting” of the first images. There is also a limit on how much information the Workspace and the Global Workspace can hold, although this limit has not been quantitatively determined yet).

5) Increasing T2 salience/arousal attenuates the AB. Increased T2 bottom-up salience leads to a higher activation of the PAM node representing T2 and thus to a target coalition with a higher activation, which increases the probability that T2 wins the competition for consciousness. In the case of emotional content with high arousal (Anderson 2005), a PAM node representing this emotion (with an activation value corresponding to the arousal) would be included in the coalition along with the target representation (Franklin et al., in press), increasing its activation and the probability of its conscious broadcast.

6) Task-irrelevant cognitive load attenuates the AB. Subject less focused on a task exhibit higher levels of tonic LC activity (see lowest panel of Fig 2), which can explain this phenomenon. In experimental conditions in which moving dots are presented around the target, and in conditions where the subject is instructed to think about something else, subjects are less focused on the AB task – therefore their AttentionCodelet Module Activation

(modeling LC activity) is higher before a target, and there is a much smaller post-target activation drop, which makes it possible to almost always report T2 accurately – in accordance with behavioral AB experiments with distracted subjects (Olivers & Nieuwenhuis 2005), (Arend et al. 2006), (Nieuwenhuis et al. 2005).

7) Target Confusion. Targets presented temporally adjacently in the same cognitive cycle (e.g. during lag-1 sparing) land in the same coalition. Since coalitions do not contain ordering information, the temporal order of the two targets is unknown to the agent. This could account for the target confusion effect. However, no exact computational mechanism has been implemented yet for reproducing how human subjects “guess” (often incorrectly, see Dux & Marois 2009, Chun 1997) the first target in such a case.

8) AB without T1 masking. There is an AB effect without the T1 mask, since the AB in this model is due to a) the refractory-like period of the ACM and b) intrusion of the post-T2 distractor, and doesn’t depend on the post-T1 distractor.

The above explanations show that our model of the AB is capable of explaining more than just the basic AB paradigm simulated in this preliminary report. To the authors’ knowledge, no model provides detailed explanations for every AB effect described above. Our LIDA based model could be able to do so, provided further computational simulations that can verify these results; and is more general than most AB models since it grows out of a universal model of cognition instead of being specific to the AB like most AB models (except for the Threaded Cognition model, which is based on ACT-R). Of the models published in the last five years, the following have the most explanatory power (Dux & Marois 2009): eSTST (Bowman & Wyble 2007), Attention Cascade (Shih 2008), Threaded Cognition (Taatgen et al. 2009), and Boost and Bounce (Olivers & Meeter 2008). Table 1 shows a comparison of these models and the proposed LIDA-based model. They are similar to our model in that they also rely on a suppressed/delayed attentional enhancement of T2; the major differences, apart from the LIDA architecture’s generality and plausibility (Baars & Franklin 2009), are the following. First, apart from a depleted attentional resource (ACM activation), which our model shares with these models (except for the Threaded Cognition model, which instead of a depleted resource relies on an unnecessary consolidation protection rule) there is also competition with the post-T2 distractor in the Global Workspace. Second, the postulated theoretical reason for T2 not being reported in some short lag trials is that the T2 percept does not win the competition for consciousness, i.e. the gamma-coherent neuronal ensembles representing T2 do not become part of the large-scale theta-gamma synchronized network representing conscious contents, which is consistent with the recently implicated importance of oscillatory activity in the AB (Janson & Kranczioch 2011). This theta-gamma oscillatory synchrony is proposed to be the neuronal basis of functional consciousness, and of the global broadcast in the LIDA model (Madl et al. 2011).

↓ AB-related phenomenon	LIDA	eSTST Bowman & Wyble 2007	Attention Cascade Shih 2008	Threaded cognition Taatgen et al. 2009	Boost and Bounce Olivers & Meeter 2008
Lag-1 sparing and spread lag-1 sparing	✓	✓	✓	✓	✓
Posttarget intrusion	✓	✓	✓	✓	✓
Whole report attenuates the AB	✓	✓	✗	✓	✗
Increased T2 salience attenuates the AB	✓	✓	✓	✓	✓
Increased T2 arousal attenuates the AB	✓	✗	✗	✗	✗
Task-irrelevant cognitive load attenuates the AB	✓	✗	✗	✓	✗
Target confusion	✓	✓	✓	✓*	✓
AB without T1 masking	✓	✓	✓	✓	✗

Table 1. A comparison of the LIDA-based AB model, and other models, conceptually accounting for AB phenomena. *: The original ACT-R based implementation does not account for target confusion, but the authors did include a simulation of target confusion using a custom visual module

The model has been implemented computationally to reproduce an experiment similar to (Potter et al. 2010) (Fig 1) to show that it can model human behavior. Every 107ms an image is presented to the agent, and it has to report two targets (images of vehicles) in a stream of distractors (faces, for easy discrimination; see Fig 3A top. The images were taken from the Caltech image database. The second target succeeds the first either immediately after 107ms (lag 1), or after a distractor (213ms, lag 2), or after 3 distractors (427ms, lag 4). Human reporting accuracies in such a setting are displayed in Fig 4 (Potter et al. 2010).

The LIDA Attentional Blink Agent is based on the LIDA computational framework (Snaider et al. 2011). Its environment consists of the screen displaying the images, and of three buttons for each possible response (first target, second target, distractor). Images are recognized using a number of feature detectors looking out for scale and rotation invariant features. These implemented feature detectors are based on Speeded Up Robust Features (Bay et al. 2008). The task instructions were pre-defined in the form of a TAC and DAC bringing relevant images to consciousness, and schemes in the Procedural Memory for reporting targets by pressing buttons. Fig 4 shows the LIDA AB Agent’s performance in this task, compared to human data. These result were obtained using the same framework, and the same parameters, as previous LIDA agents (Madl et al. 2011), except for the addition of the ACM activity modeling LC activation.

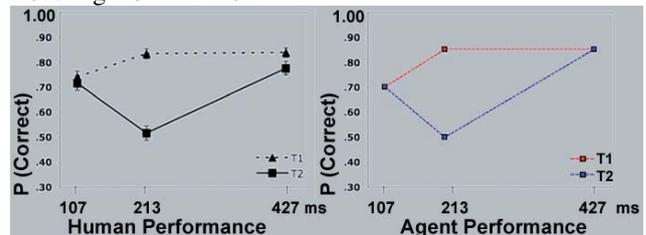


Fig 4. Comparison of human and the LIDA AB Agent’s performance in a visual AB task. Left panel adapted from (Potter et al. 2010).

References

- Anderson, A.K., 2005. Affective influences on the attentional dynamics supporting awareness. *J Exp Psychol Gen*, 134(2), p.258.
- Arend, I., Johnston, S. & Shapiro, K., 2006. Task-irrelevant visual motion and flicker attenuate the attentional blink. *Psychon Bul Rev*, 13(4), pp.600–607.
- Aston-Jones, G. & Cohen, J.D., 2005. An integrative theory of locus coeruleus-norepinephrine function: adaptive gain and optimal performance. *Annu Rev Neurosci*, 28, pp.403–450.
- Aston-Jones, G., Rajkowski, J. & Cohen, J., 1999. Role of locus coeruleus in attention and behavioral flexibility. *Biol Psych*, 46(9), pp.1309–1320.
- Baars, B.J., 1988. *A cognitive theory of consciousness*, Cambridge Univ Pr.
- Baars, B.J. & Franklin, S., 2009. Consciousness is computational: The LIDA model of global workspace theory.
- Barsalou, L.W., 1999. Perceptual symbol systems. *Behav Brain Sci*, 22(04), pp.577–660.
- Bay, H. et al., 2008. Speeded-up robust features (SURF). *Comput Vis Image Und*, 110(3), pp.346–359.
- Bichot, N.P. & Schall, J.D., 1999. Effects of similarity and history on neural mechanisms of visual selection. *Nat Neurosci*, 2, pp.549–554.
- Bowman, H. & Wyble, B., 2007. The simultaneous type, serial token model of temporal attention and working memory. *Psychol Rev*, 114(1), p.38.
- Chun, M.M., 1997. Temporal binding errors are redistributed by the attentional blink. *Atten Percept Psychophys* 59(8), pp.1191–1199.
- Doesburg, S.M. et al., 2009. Rhythms of Consciousness: Binocular Rivalry Reveals Large-Scale Oscillatory Network Dynamics.
- Duncan, J., Humphreys, G. & Ward, R., 1997. Competitive brain activity in visual attention. *Curr Opin Neurobiol*, 7(2), pp.255–261.
- Dux, P.E. & Marois, R., 2009. The attentional blink: A review of data and theory. *Atten Percept Psychophys*, 71(8), pp.1683–1700.
- Franklin, S. et al., in press. LIDA: A Systems-level Architecture for Cognition, Emotion, and Learning. *BICA*.
- Giesbrecht, B. & Di Lollo, V., 1998. Beyond the attentional blink: Visual masking by object substitution. *J Exp Psychol Human*, 24(5), p.1454.
- Hofstadter, D.R. & Mitchell, M., 1994. The Copycat Project: A model of mental fluidity and analogy-making. *Adv. in connectionist and neural comp theory*, 2, pp.31–112.
- Janson, J. & Kranczioch, C., 2011. Good vibrations, bad vibrations: Oscillatory brain activity in the attentional blink. *Adv Cog Psych*, 7(-1), pp.92–107.
- Knudsen, E.I., 2007. Fundamental components of attention. *Annu. Rev. Neurosci.*, 30, pp.57–78.
- Lavie, N. & Cox, S., 1997. On the efficiency of visual selective attention: Efficient visual search leads to inefficient distractor rejection. *Psychol Sci*, 8(5), p.395.
- Liu, H. et al., 2009. Timing, timing, timing: Fast decoding of object information from intracranial field potentials in human visual cortex. *Neuron*, 62(2), pp.281–290.
- Madl, T., Baars, B.J. & Franklin, S., 2011. The Timing of the Cognitive Cycle. *PLoS ONE*, 6(4), p.e14803.
- Marois, R., Yi, D.J. & Chun, M.M., 2004. The neural fate of consciously perceived and missed events in the attentional blink. *Neuron*, 41(3), pp.465–472.
- Martens, S. & Wyble, B., 2010. The attentional blink: past, present, and future of a blind spot in perceptual awareness. *Neurosci Biobehav R*, 34(6), pp.947–957.
- Minsky, M., 1988. *The society of mind*, Simon and Schuster.
- Nieuwenhuis, S. et al., 2005. The role of the locus coeruleus in mediating the attentional blink: A neurocomputational theory. *J Exp Psychol G*, 134(3), p.291.
- Nieuwenstein, M.R., Potter, M.C. & Theeuwes, J., 2009. Unmasking the attentional blink. *J Exp Psychol Human* 35(1), p.159.
- Olivers, C.N. & Meeter, M., 2008. A boost and bounce theory of temporal attention. *Psychol Rev*, 115(4), p.836.
- Olivers, C.N. & Nieuwenhuis, S., 2005. The beneficial effect of concurrent task-irrelevant mental activity on temporal attention. *Psychol Sci*, 16(4), p.265.
- Olivers, C.N., van der Stigchel, S. & Hulleman, J., 2007. Spreading the sparing: Against a limited-capacity account of the attentional blink. *Psychol Res*, 71(2), pp.126–139.
- Potter, M.C. et al., 2010. Picture detection in rapid serial visual presentation: Features or identity? *J Exp Psych Human*, 36(6), p.1486.
- Potter, M.C., Nieuwenstein, M. & Strohminger, N., 2008. Whole report versus partial report in RSVP sentences. *J Mem Lang*, 58(4), pp.907–915.
- Sara, S.J., 2009. The locus coeruleus and noradrenergic modulation of cognition. *Nat Rev Neurosci*, 10(3), pp.211–223.
- Serences, J.T. & Yantis, S., 2006. Selective visual attention and perceptual coherence. *Trends in Cognitive Sciences*, 10(1), pp.38–45.
- Shih, S.I., 2008. The attention cascade model and attentional blink. *Cog Psychol*, 56(3), pp.210–236.
- Snider, J., McCall, R. & Franklin, S., 2011. The LIDA framework as a general tool for AGI. *Artificial General Intelligence*, pp.133–142.
- Taatgen, N.A. et al., 2009. Too much control can hurt: A threaded cognition model of the attentional blink. *Cog Psychol*, 59(1), pp.1–29.
- Thompson, K.G., Biscoe, K.L. & Sato, T.R., 2005. Neuronal Basis of Covert Spatial Attention in the Frontal Eye Field. *J Neurosci*, 25(41), pp.9479–9487.
- Warren, C.M. et al., 2009. Target—distractor interference in the attentional blink implicates the locus coeruleus—norepinephrine system. *Psychon Bul Rev*, 16(6), pp.1106–1111.
- Wyble, B., Bowman, H. & Nieuwenstein, M., 2009. The attentional blink provides episodic distinctiveness: Sparing at a cost. *J Exp Psych Human*, 35(3), p.787.

Self-Organization and Associative Learning as a Basis for Cognitive and Sensorimotor Modeling of Speech Production, Perception, and Acquisition

Bernd J. Kröger, Cornelia Eckers, Christiaine Neuschaefer-Rube
({bkroeger, ceckers, cneuschaefer}@ukaachen.de)

Department of Phoniatrics, Pedaudiology, and Communication Disorders, RWTH Aachen University,
Pauwelsstr. 30, 52074 Aachen, Germany

Keywords: sensorimotor modeling; cognitive modeling; speech production; speech perception; speech acquisition.

The Structure of the Model

A computational model has been proposed which is capable of simulating early phases of speech acquisition, speech production, and speech perception. The model comprises two main modules, i.e. mental lexicon and action repository (Fig. 1). The mental lexicon activates semantic and phonological representations of words (cognitive level, Li et al. 2004) while the action repository activates sensory and motor representations of syllables (cf. Levelt & Wheeldon 1994, Guenther et al. 2006, Kröger et al. 2009, Kröger et al. 2011a).

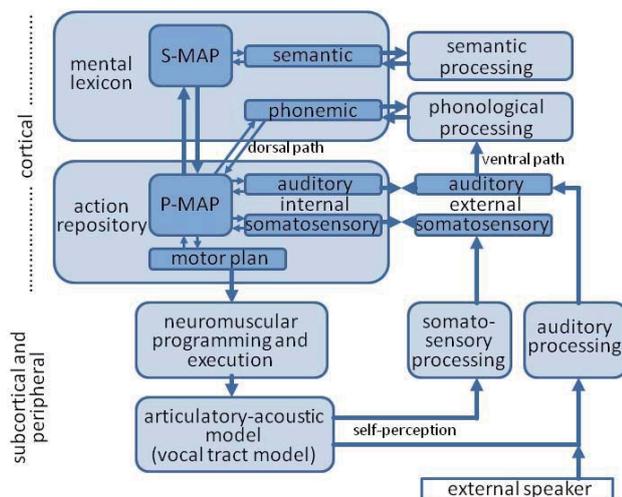


Figure 1: Structure of our cognitive-sensorimotor model of speech processing. Light blue boxes indicate processing modules; dark blue boxes indicate self-organizing maps (i.e. semantic map S-MAP and phonetic map P-MAP) or neural state maps, i.e. the semantic, phonemic, auditory, somatosensory, and motor plan state map (for a detailed description of this model see Kröger et al. 2011a).

The sensorimotor part of our model has been implemented and tested by simulating early phases of speech acquisition (i.e. babbling phase and imitation phase) and performing production and perception tests after learning (Kröger et al. 2009 and 2011b). The detailed structure of the sensorimotor part of our model is given in Fig. 2 (top). Cortical regions associated with specific neural maps within our approach, are displayed in Fig. 2 (bottom).

A characteristic feature of our approach is that we assume two self-organizing maps – i.e. a semantic map (S-MAP, Fig. 1) and a phonetic map (P-MAP, Fig. 1) – which on the one hand are associated with each other and which on the other hand are associated with state maps, representing current semantic, motor plan or sensory activation patterns of speech items (cf. Li et al. 2004, Zhao et al. 2011).

Acquisition of Skills and Knowledge

Acquisition is simulated in our approach by applying a huge amount of training items to the model. These training items represent stimuli, which are exposed to a newborn and later on to a toddler (i.e. to the model) during the first two years of lifetime.

Acquisition starts with “babbling”, i.e. a training phase which is mainly language independent. Here the model generates random motor patterns (motor plan states) and produces appropriate auditory and somatosensory patterns (auditory and somatosensory states). Motor plan and sensory states are exposed to the model nearly simultaneously and thus allow associative learning, i.e. an association of specific motor plan states with corresponding sensory states (Kröger et al. 2009). This learning leads to an adjustment of neural connections between state maps and the self-organizing phonetic map (P-MAP). Neurons within the phonetic map represent specific sensorimotor states and these states are ordered with respect to phonetic features within this map (i.e. self-organization). Thus, after learning co-activation of a motor plan state is possible, if a specific sensory (e.g. auditory) state is activated. In this way, initial sensorimotor knowledge is acquired for “proto-vowels” and “proto-CV-syllables” (the term “proto” refers to the fact that these phonetic states are not necessarily language specific; cf. Kröger et al. 2009).

This initial sensorimotor knowledge later on allows “imitation training” since after initial babbling the model is able to imitate external auditory stimuli. Imitation training leads to a further adjustment of neural connections between P-MAP and state maps and leads to a further ordering of states within the P-MAP, which now leads to language-specific speaking skills (Kröger et al. 2011b). Beside further development of the action repository, imitation training is also the starting point for building up the mental lexicon (ibid.). Training items for imitation training comprise sensorimotor states, which result from imitation trials performed by the model itself, but in addition comprise a semantic representation of the word which is currently imitated. This allows a

parallel self-organization of the S-MAP and adjusts the neural connections between both self-organizing maps (P-MAP and S-Map) as well as the neural connections to all state maps. Thus, imitation training in addition leads to the formation of first language specific knowledge (i.e. phonological representation of words, see Kröger et al. 2011a).

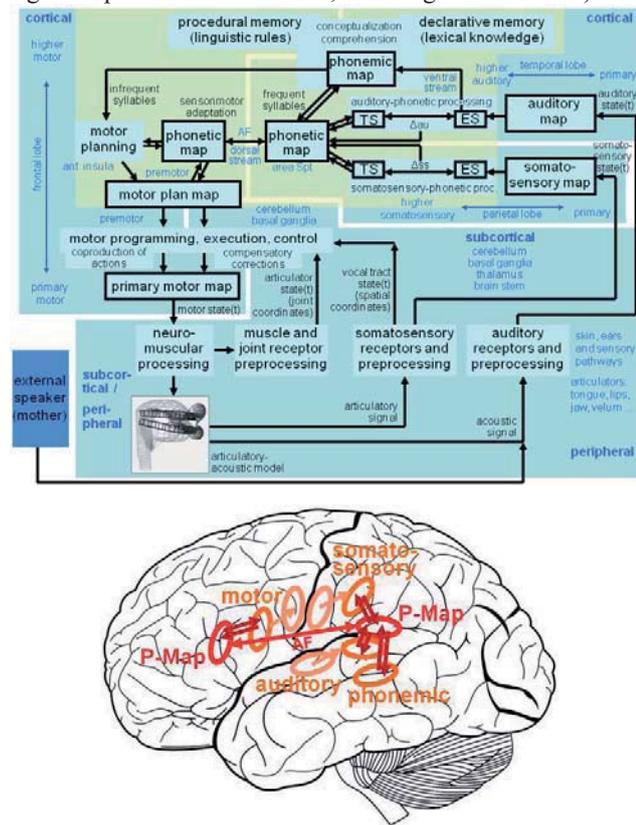


Figure 2: Structure of the sensorimotor part of the model (top) and cortical regions related to specific neural maps occurring within the model (bottom). **Top:** The model comprises a feedforward pathway (motor) and three feedback pathways (lower and higher level somatosensory and auditory). Outlined boxes indicate neural maps; other boxes indicate neural processing modules, which are not specified in detail. Single arrows indicate neural pathways for forwarding information; double arrows indicate neural mappings which are involved in information processing. The light green area indicates higher processing levels which activate syllables as entire units; lower levels (primary cortical maps and subcortical structures) are capable of processing smaller temporal units of production and perception. TS: map for trained sensory states (already acquired); ES: map for external sensory states (currently produced); Δu : auditory error signal; Δs : somatosensory error signal. **Bottom:** Cortical locations of implementations of the primary neural maps (pink: primary motor map, primary somatosensory map, and primary auditory map), of the neural state maps (orange: phonemic map, motor plan map, somatosensory and auditory state maps within the somatosensory-phonetic and auditory-phonetic processing modules), and of two

mirrored representations of the phonemic map (red). Neural mappings between state maps and phonemic map are indicated by dark red arrows; The neural pathway between the two mirrored representations of the phonemic map is indicated by a red arrow; AF: arcuate fasciculus.

Future Work

The phonemic as well as the semantic map are the central maps for self-organization and enable associative learning in our approach. While the gross structure of our model is in accordance with the well-known models introduced by Guenther et al. (2006) and by Li et al. (2004), the reality of the phonemic map (e.g. in its mirrored location, joined via the AF, as postulated in Fig. 2) needs to be proved by brain imaging experiments.

Acknowledgments

This work was supported in part by German Research Council (DFG) grants Kr 1439/13-1 and Kr 1439/15-1.

References

- Guenther, F.H., Ghosh, S.S. & Tourville, J.A. (2006). Neural modeling and imaging of the cortical interactions underlying syllable production. *Brain and Language*, 96, 280-301.
- Kröger, B.J., Kannampuzha, J. & Neuschaefer-Rube, C. (2009). Towards a neurocomputational model of speech production and perception. *Speech Communication*, 51, 793-809.
- Kröger, B.J., Birkholz, P. & Neuschaefer-Rube, C. (2011a). Towards an articulation-based developmental robotics approach for word processing in face-to-face communication. *PALADYN Journal of Behavioral Robotics*, 2, 82-93.
- Kröger, B.J., Birkholz, P., Kannampuzha, J., Kaufmann, E. & Neuschaefer-Rube, C. (2011b). Towards the acquisition of a sensorimotor vocal tract action repository within a neural model of speech processing. In A. Esposito, A. Vinciarelli, K. Vicsi, C. Pelachaud & A. Nijholt (Eds.), *Analysis of Verbal and Nonverbal Communication and Enactment: The Processing Issues. LNCS 6800*. Berlin: Springer, pp. 287-293.
- Levelt, W.J.M. & Wheeldon, L. (1994). Do speakers have access to a mental syllabary? *Cognition*, 50, 239-269.
- Li, P., Farkas, I. & MacWhinney, B. (2004). Early lexical development in a self-organizing neural network. *Neural Networks*, 17, 1345-1362.
- Zhao, X., Li, P. & Kohonen, T. (2011). Contextual self-organizing map: software for constructing semantic representations. *Behavior Research Methods* 43, 77-88.

The Role of Memory in MHP/RT: Organization, Function and Operation

Muneo Kitajima (mkitajima@kjs.nagaokaut.ac.jp)

Nagaoka University of Technology
1603-1 Kamitomioka Nagaoka, Niigata 940-2188 Japan

Makoto Toyota (pubmtoyota@me.com)

T-Method
Chiba 264-0032 Japan

Abstract

To develop a unified theory of human decision-making in daily behavior selections, the authors propose an architecture model called Model Human Processor with Real Time constraints (MHP/RT) (Kitajima & Toyota, 2012). This model integrates the established theory of decision-making by Kahneman (2003), Two Minds, and the idea that human behavior is organized in the ever-changing environment (Newell, 1990) into a construct that is capable of simulating such daily behavior as driving a car or watching a baseball game at a stadium. Kitajima and Toyota (2012) proposed that MHP/RT operates in one of four modes that are defined by the active components of MHP/RT at a specific time. Kitajima and Toyota (2011a) demonstrated that at a specific moment MHP/RT is processing one of four aspects of a certain event. This paper demonstrates how memory is used in the four operation modes and the four processing modes of MHP/RT.

Keywords: decision making; behavior selection; Two Minds; time scale of action; MHP/RT; autonomous memory;

Introduction

Traditionally, human behavior is considered as the outcome of conscious and unconscious processes, which involve conscious and unconscious operations using necessary pieces of information from long-term memory in which experiences are stored in representations accessible to these processes. From this perspective, the role of memory is similar to a database system. It stores a huge amount of data to be used on request by other systems that work to accomplish some goals.

However, an increasing voice suggests that the memory system is better viewed as an autonomous system, rather than a passive database system. For example, Marcus (2008) wrote:

Nobody knows for sure how this [memory] works, but my best guess is that each of our brain's memories acts *autonomously*, on its own, in response to whatever requests it might match, thereby eliminating the need for a central agent to keep a map of memory storage locations. Of course, when you rely on matches rather than specific locations that are known in advance, there's no guarantee that the *right memory* will respond; the fewer the cues you provide, the more "hits" your memory will serve up, and as a consequence the memory that you actually want may get buried among those that you don't want. – *adapted from pp.22–23 of Marcus (2008)*

The authors (Kitajima & Toyota, 2012) developed an architecture model capable of simulating human behavior selec-

tions in real-world situations. The basic idea is that observed human behavior is the result of synchronized integration of the output of conscious and unconscious processes, with the support of the memory system which works autonomously and information in long-term memory becomes available by means of resonance processes, not by retrieval processes initiated by either the conscious or unconscious process.

The architecture model is an integration of two established principles of human behavior: 1) Two Minds, which refers to conscious and unconscious processes that work in decision-making, proposed by Kahneman (2003); J. S. B. Evans (2003) and 2) the time scale of human action suggested by Newell (1990), which regards conscious processes as very slow feedback processes and unconscious processes as very fast feed-forward processes (see the next section for brief descriptions of these principles). Kitajima and Toyota (2012) describes the architecture model, Model Human Processor with Real-Time constraints (MHP/RT), as integrating Two Minds and Newell's time scale of action with special consideration of how to synchronize these two totally different systems in terms of their characteristic times. We also demonstrated that this model can plausibly simulate passengers' behaviors at train stations (*e.g.*, transferring to another line, using the toilet, and purchasing train tickets). Kitajima and Toyota (2011a) demonstrated that for a certain behavioral event $Event(T)$ that happens at a certain time T , MHP/RT addresses this event in four different ways, or modes, that occur serially. In other words, human behavior is considered to be a series of these four different modes: conscious or unconscious processes concerning $Event(T)$ before it happens ($t < T$) or after it happens ($t > T$).

MHP/RT defines memory as an autonomous system. However, previous publications (Kitajima & Toyota, 2011a, 2012) have not described in detail how memory is used. The purpose of this paper is to fill this gap by demonstrating the operation of the four processing modes of MHP/RT from the viewpoint of the role of memory. This paper starts by briefly describing MHP/RT (see Kitajima and Toyota (2012) for more detail), then discussing the four operation modes of MHP/RT (see Kitajima and Toyota (2011a) for more detail), and finally describing the role of memory in the four processing modes of MHP/RT.

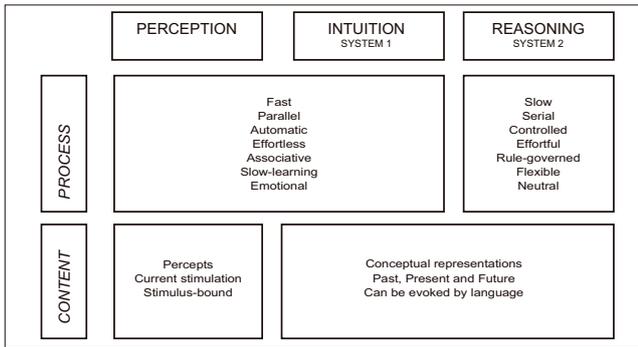


Figure 1: Two Minds (Kahneman, 2003).

The Principles for Understanding Human Behavioral Selections

In this section, we will review briefly Kahneman's Two Minds (Kahneman, 2003) and Newell's time scale of human action (Newell, 1990).

Two Minds: The Theory of Decision-Making

Human decision-making has been a central topic in economics. Herbert A. Simon, winner of the Nobel Prize in economics in 1978, proposed principles of human beings' decision-making processes. He described the decision-making process as a "bounded rationality principle" as well as a "satisficing principle" (Simon, 1956, 1996). Simon claimed that agents, or human beings, face uncertainty about the future and costs when acquiring information in the present. These factors limit the extent to which human beings can make a fully rational decision. Thus, they possess only "bounded rationality" and must make decisions by "satisficing," or choosing the path that might not be optimal, but which will make them happy enough.

Recently, Kahneman, winner of the Nobel Prize in economics in 2002, introduced behavioral economics, which stems from the claim that decision-making is governed by the so-called "Two Minds" (Kahneman, 2003; J. S. B. T. Evans & Frankish, 2009). In other words, a human being's behavior is the outcome of two different systems including an "experiential processing system (System 1)" and a "rational processing system (System 2)." Figure 1, adapted from (Kahneman, 2003), illustrates the workings of the two systems. In short, System 1 is a fast feed-forward control process driven by the cerebellum and oriented toward immediate action. In contrast, System 2 is a slow feedback control process driven by the cerebrum and oriented toward future action.

Newell's Time Scale of Human Action

The two systems, System 1 and System 2, work jointly and in synchronous with the ever-changing external world to exhibit moment by moment coherent human behavior. However, there is a large difference in processing speed between the two systems. Rational processing typically takes minutes

Time Scale of Human Action			
Scale (sec)	Time Units	System	World (Theory)
10^7	months		
10^6	weeks		Social Band
10^5	days		
10^4	hours	Task	
10^3	10 min	Task	Rational Band
10^2	minutes	Task	
10^1	10 sec	Unit Task	
10^0	1 sec	Operations	Cognitive Band
10^{-1}	100 ms	Deliberate Act	
10^{-2}	10 ms	Neural Circuit	
10^{-3}	1 ms	Neuron	Biological Band
10^{-4}	100 μ s	Organelle	

Figure 2: Newell's time scale of human action (Newell, 1990).

to hours whereas experiential processing typically extends from hundreds of milliseconds to tens of seconds. Figure 2 illustrates the time scale of human action consisting of the following four bands, 1) Biological Band, 2) Cognitive Band, 3) Rational Band, and 4) Social Band, each has its characteristic processing time (Newell, 1990). A large part of human beings' daily activities are immediate actions and are therefore under control of the experiential processing system (System 1). The rational processing system (System 2) intervenes with the experiential processing system to better organize the overall outcome of the processing through consciously envisioning possible futures.

MHP/RT: Integration of MHP and Two Minds Brief Description of MHP/RT

Toyota and Kitajima (2010a) and Kitajima and Toyota (2012) proposed MHP/RT as a simulation model of human behavior selection¹. It stems from the successful simulation model of human information processing, Model Human Processor (MHP) (Card, Moran, & Newell, 1983), and extends it by incorporating three theories we have published in the cognitive sciences community. The Maximum Satisfaction Architecture (MSA) deals with coordination of behavioral goals (Kitajima, Shimada, & Toyota, 2007), the Structured Meme Theory (SMT) involves utilization of long-term memory, which works as an autonomous system (Toyota, Kitajima, & Shimada, 2008), and Brain Information Hydrodynamics (BIH) involves a mechanism for synchronizing the individual with the environment (Kitajima, Toyota, & Shimada, 2008).

¹Unfortunately, the detailed description of the model is available only in Japanese in Kitajima and Naito (2010) and Kitajima and Toyota (2011b).

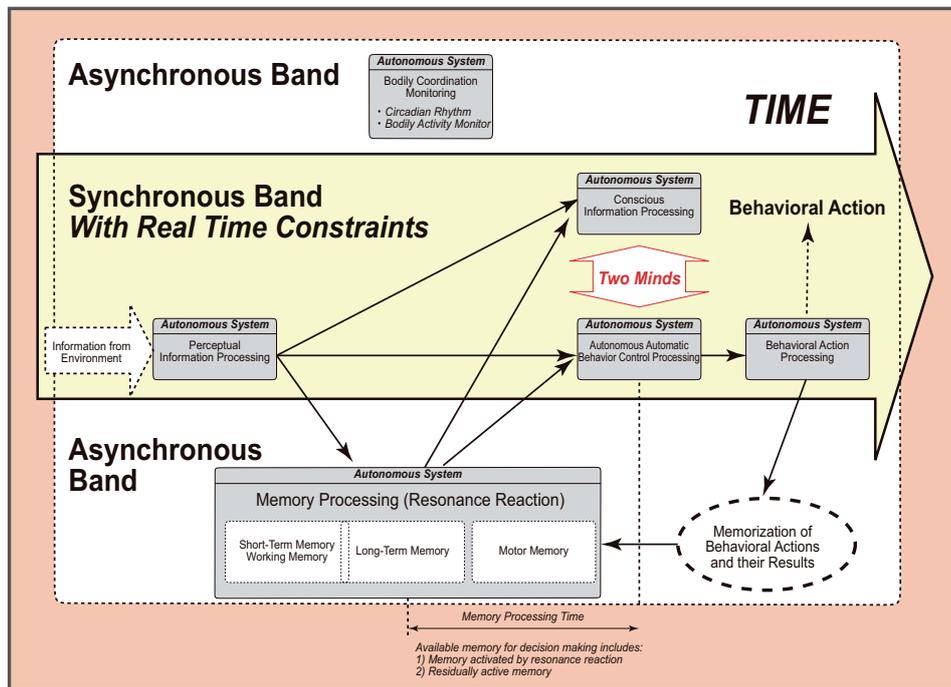


Figure 3: Schematic diagram of MHP/RT (Kitajima & Toyota, 2012).

MHP/RT includes a mechanism for synchronizing autonomous systems (rectangles with rounded corners in Figure 3), working in the “Synchronous Band.” MHP/RT was created by combining MHP and Two Minds by applying our conceptual framework of Organic Self-Consistent Field Theory (Toyota & Kitajima, 2010b). See Kitajima (2011) for more information.)

MHP/RT works as follows:

1. Inputting information from the environment and the individual,
2. Building a cognitive frame in working memory, which is not depicted in the figure but it resides between the conscious process and the unconscious process to interface them,
3. Resonating the cognitive frame with autonomous long-term memory to make available the relevant information stored in long-term memory; cognitive frames are updated at a certain rate and the contents in the cognitive frames are continuously input to long-term memory to make pieces of information in long-term memory accessible to System 1 and System 2,
4. Mapping the results of resonance on consciousness to form a reduced representation of the input information, and
5. Predicting future cognitive frames to coordinate input and working memory.

As depicted in Fig. 3, human beings operate in two bands, the asynchronous band and the synchronous band. The Bodily Coordination Monitoring System and the Memory Processing System operate in the asynchronous band. The Per-

ceptual Information Processing System, Conscious Information Processing System, Autonomous Automatic Behavior Control Processing System, and Behavioral Action Processing System operate in the synchronous band. These systems work autonomously. System 1 of the Two Minds corresponds to the Autonomous Automatic Behavior Control Processing System, and System 2 corresponds to the Conscious Information Processing System.

The density of information in working memory is the product of the updating rate of the cognitive frame and the degree of fineness of the information represented in the cognitive frame. When the system is under the control of automatic behavior (System 1), the updating rate of the cognitive frame tends to be high; however, the degree of fineness of the information represented in the cognitive frame is coarse. When the system is under the control of consciousness (System 2), the updating rate of the cognitive frame and the degree of fineness of the information are flexibly determined by the context.

Hierarchical structure of behavior. Observed behavior should be regarded as a compound of activities that occur on different time scales. The time scales may be milliseconds, hundreds of milliseconds, a few minutes, or even a few weeks. It is not true that activities that occur on a certain time scale evolve continuously to the next time scale. Rather, it is more appropriate to assume that a set of activities that occur on a certain time scale are discontinuously connected with higher-level activities, and therefore the relationship between a pair of related activities at two different levels is non-linear. Newell (1990) explained the time scale of human action, and

identified four bands and their characteristic times: the biological band (1msec ~ 10msec), the cognitive band (100msec ~ 10sec), the rational band (a few minutes ~ a few hours), and the social band (a few hours ~ a few hours).

Interaction between System1 and System 2. MHP/RT transforms the input information from the environment to the output behavior to the environment. The actual operation is determined by the relative balance between the following two factors, which is determined by the degree of participation of consciousness in the manifestation of behavior at each moment:

1. The effect of feedback from the conscious layer (System 2) on shaping behavior, and
2. The effect of feedforward control from the autonomous automatic behavior control layer (System 1) on shaping behavior.

How System 1 and System 2 interact appears in the relationship between the updating rate of the cognitive frame and the density of information represented in the cognitive frame. In the following, this will be explained in more detail.

- **System 1 control mode:** When System 1 governs behavior, the updating rate of the cognitive frame is the fastest, and the system behaves unconsciously. The system refers to the memory that is activated via the resonance reaction, and the outcome of behavior is consciously monitored. As long as the output of behavior is consistent with the representation of the contents of activated memory, no feedback control is applied. An example of this behavior mode is riding a bicycle on a familiar road. It is not necessary to monitor the behavior with high frequency. As a result, System 2 may initiate tasks that are not directly relevant to unconscious behavior. In such a situation, consciousness is free from behavior that is tightly embedded in the environment. Therefore, for example, the system may use a mobile phone to talk with a friend while riding a bicycle.
- **System 2 control mode:** When System 2 governs behavior, the systems try to behave according to the image System 2 created or meditate with no bodily movement. The least resources are allocated for initiating behavior according to input from the environment. This corresponds to a situation in which the amount of flow of information in System 1 is small. Working memory is occupied by activities related to System 2. However, the sensory-information filter functions so that the system can react to a sudden interruption from the environment (*e.g.*, a phone call).

How MHP/RT Works

At a given time, T , MHP/RT's state is viewed in two ways; 1) which part of MHP/RT is working and 2) what content MHP/RT is processing. In the following subsections, we describe the "which part" question in the "Four Operation Modes" subsection (Kitajima & Toyota, 2012), and the "for what" question in the "Four Processing Modes" subsection (Kitajima & Toyota, 2011a).

Four Operation Modes of MHP/RT

In MHP/RT, behavior is the outcome of activities in System 1 and System 2, both of which use working memory to prepare for the next action. Depending on the situation, behavior is driven mainly by either System 1 (MHP/RT Mode 1) or System 2 (MHP/RT Mode 2). Both systems work synchronously by sharing working memory. However, in some situations, both work asynchronously (MHP/RT Mode 3) or independently (MHP/RT Mode 4); working memory may be shared weakly or used solely for one of these layers.

Four Processing Modes of MHP/RT

Human behavior is considered a series of moment-by-moment decision-making processes in the ever-changing environment. Each decision-making process is carried out by System 1 and System 2 of Two Minds under real-time constraints, which basically requires synchronizing the workings of System 1 and System 2 in the real world by taking into account each system's characteristic times defined by Newell's time scale of action (Fig. 2). The result of decision-making is an event that includes the direct output of decision-making or behavior, and the resultant state of the external world.

The four processing modes in human decision-making are: (1) conscious (System 2) behavior before the event, (2) conscious (System 2) behavior after the event, (3) unconscious (System 1) behavior before the event, and (4) unconscious (System 1) behavior after the event.

Figure 4 illustrates these four processing modes along the time dimension expanding before and after the event, which is denoted as the "boundary event" in the figure.

The Role of Memory in MHP/RT

Organization

As Figure 3 illustrates, the memory system operates *asynchronously* with the systems working synchronously with the environment. Memory processes include the storage of information and the use of stored information, which play a very important role in the real-time simulation of human decision-making in daily life.

Memory storage. We assume that memory is organized by a "Multi-Dimensional Frame" (MD frame) for storing information. The MD frame is a conceptual extension of Minsky's frame (Minsky, 1988). It is a primitive cognitive unit that conveys information that the brain can manipulate under various constraints, similar to the concept of the Idealized Cognitive Model (ICM) theory by Lakoff (1987) and the schema theory by Rumelhart (1980). Our theory involves two kinds of MD frame. The Behavior Multidimensional frame (BMD frame) is created and used by Autonomous Automatic Behavior Control Processing. The Relational Multidimensional frame (RMD frame) is created and used by Conscious Processing. The BMD frame and RMD frame are connected by a sharing Object originating from Perceptual Processing.

Due to the limitation of the brain's processing capability, the range of integration is limited; therefore, System 1

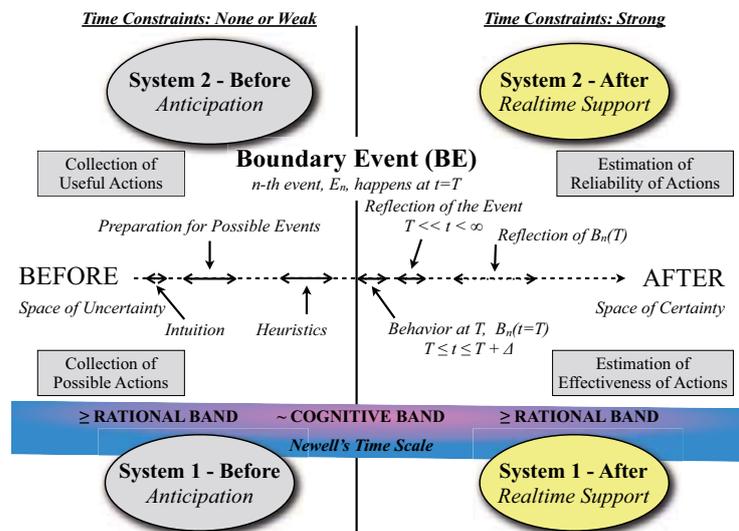


Figure 4: How the Four Processing Modes work (Kitajima & Toyota, 2011a).

memory and System 2 memory may differ. However, they may share objects originating from perceptual sensors. Thus, when objects that are the result of the just-finished integration and segmentation process are processed in the next cycle, representation of the objects may serve as common elements to combine System 1 memory and System 2 memory to form an intersystem memory. We call this memory the Multidimensional (MD) Frame.

Memory usage. Object cognition involves collecting information from the environment via perceptual sensors; integrating and segmenting the collected information, centering on visually collected objects; and continuing these processes until the objects necessary to live in the environment are obtained. These objects are then used independently in System 1 and System 2 of Two Minds, and memorized after integrating related entities associated with each system.

Function: Resonance

At a given moment, MHP/RT is working in one of the four operation modes described above. However, the memory system works autonomously to make part of long-term memory active so that it can be used in System 1 and/or System 2 processing through resonance processes. However, as depicted in Figure 5, how the memory system reacts to the environment may depend on the degree of time constraints that the human-environment system imposes on itself. When real-time constraints are strong, slow memory processes that use long-term memory do not participate in the processing. In other words, only the unconscious side of the Two Minds system, System 1, works and has a chance to use memory through resonance. In contrast, with few real-time constraints, the conscious and unconscious systems work collaboratively in some cases and independently in other cases. Both systems have a chance to use as many resonated contents as possible.

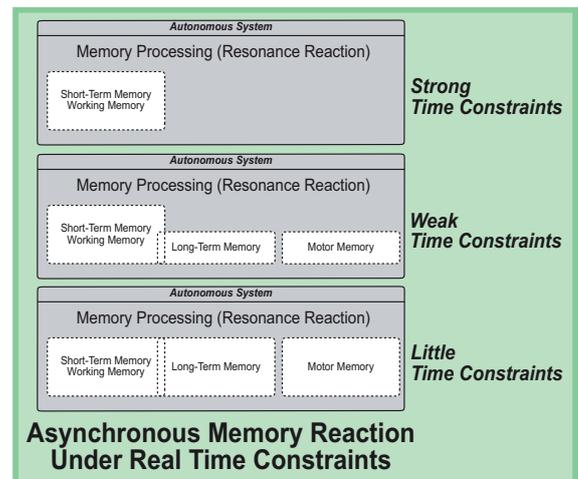


Figure 5: Memory reaction under real time constraints.

Operation: Pipelining

At a given moment, MHP/RT is processing one of four content types: a future event consciously or unconsciously, or a past event consciously or unconsciously. For future/conscious processing, MHP/RT uses memory that conveys a sequence of actions with symbolic representations for accomplishing a currently held goal. For future/unconscious processing, it uses memory that is associated with an automatic sequence of actions that should lead to the goal. For past/conscious processing, it reflects on and elaborates a certain symbolic event by using activated pieces of knowledge through resonance processes. For past/unconscious processing, existing memory is modified by using activated non-symbolic pieces of knowledge that is currently activated in working memory.

It is important to note that memory activation is a totally parallel process; therefore, there is no way of knowing which part of activated memory is used. It depends completely on which object MHP/RT is processing. MHP/RT's resonance process makes available the relevant part of activated knowledge through resonance. Along the time dimension, MHP/RT, working in one of four operation modes, switches among the four processing modes and uses activated knowledge through resonance. MHP/RT's processing is a pipeline process of four primitive processes. The nature of this pipelining may change depending on the nature of the task. When learning a new task, it is impossible to foresee the future; therefore, past/conscious processing may dominate. In contrast, for example, when an experienced piano player is playing a well-practiced tune, future/unconscious processing may dominate.

Conclusion

This paper demonstrated the role of memory in MHP/RT, the architecture model of human behavior selection. The purpose of MHP/RT is to simulate human behavior; therefore, the organization, function, and operation of memory were specified accordingly. According to the specification of MHP/RT in Fig. 3, the organization of memory is defined as the MD frame. The content in long-term memory is made available through resonance processes in MHP/RT. Given that MHP/RT works in one of four different operation modes and that it processes contents associated with an event in one of four different ways, the portion of activated memory that is used may differ. We believe MHP/RT with an autonomous memory system is capable of simulating human behavior in real-world settings.

References

- Card, S. K., Moran, T. P., & Newell, A. (1983). *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Evans, J. S. B. (2003). In two minds: dual-process accounts of reasoning. *Trends in Cognitive Sciences*, 7(10), 454–459.
- Evans, J. S. B. T., & Frankish, K. (Eds.). (2009). *In Two Minds: Dual Processes and Beyond*. Oxford: Oxford University Press.
- Kahneman, D. (2003). A perspective on judgment and choice. *American Psychologist*, 58(9), 697–720.
- Kitajima, M. (2011). *Organic Self-Consistent Field Theory*. <http://kjs.nagaokaut.ac.jp/mkitajima/organic-self-consistent-field-theory/index.html>.
- Kitajima, M., & Naito, K. (Eds.). (2010). *The Science of Consumer Behavior*. Tokyo: Tokyo Denki University Press. (in Japanese)
- Kitajima, M., Shimada, H., & Toyota, M. (2007). MSA:Maximum Satisfaction Architecture – a basis for designing intelligent autonomous agents on web 2.0. In D. S. McNamara & J. G. Trafton (Eds.), *Proceedings of the 29th Annual Conference of the Cognitive Science Society* (p. 1790). Austin, TX: Cognitive Science Society.
- Kitajima, M., & Toyota, M. (2011a). Four Processing Modes of *in situ* Human Behavior. In A. V. Samsonovich & K. R. Jóhannsdóttir (Eds.), *Biologically Inspired Cognitive Architectures 2011 - Proceedings of the Second Annual Meeting of the BICA Society* (pp. 194–199). Amsterdam, The Netherlands: IOS Press.
- Kitajima, M., & Toyota, M. (2011b). *Practical Guide to CCE (Cognitive Chrono-Ethnography) – A Study Methodology of Human Beings' Behavioral Ecology Based on Cognitive Science*. Tokyo: ON-BOOK.ltd. (in Japanese)
- Kitajima, M., & Toyota, M. (2012). Simulating navigation behaviour based on the architecture model Model Human Processor with Real-Time Constraints (MHP/RT). *Behaviour & Information Technology*, 31(1), 41–58.
- Kitajima, M., Toyota, M., & Shimada, H. (2008). The Model Brain: Brain Information Hydrodynamics (BIH). In B. C. Love, K. McRae, & V. M. Sloutsky (Eds.), *Proceedings of the 30th Annual Conference of the Cognitive Science Society* (p. 1453). Austin, TX: Cognitive Science Society.
- Lakoff, G. (1987). Concepts and Conceptual Development: Ecological and Intellectual Factors in Categorization. In U. Neisser (Ed.), (pp. 63–100). New York, NY: Cambridge University Press.
- Marcus, G. (2008). *Kluge: The haphazard construction of the human mind*. Boston, MA: Houghton Mifflin Company.
- Minsky, M. (1988). *Society of Mind*. New York City, NY: Simon & Schuster.
- Newell, A. (1990). *Unified Theories of Cognition (The William James Lectures, 1987)*. Cambridge, MA: Harvard University Press.
- Rumelhart, D. E. (1980). Theoretical Issues in Reading Comprehension. In R. J. Spiro, B. C. Bruce, & W. F. Brewer (Eds.), (pp. 35–58). Lawrence Erlbaum, NJ: Cambridge University Press.
- Simon, H. A. (1956). Rational choice and the structure of the environment. *Psychological Review*, 63, 129–138.
- Simon, H. A. (1996). *The Sciences of the Artificial* (Third ed.). Cambridge, MA: The MIT Press.
- Toyota, M., & Kitajima, M. (2010a). MHP/RT: Model Human Processor with Real Time Constraints. In S. Ohlsson & R. Catrambone (Eds.), *Proceedings of the 32nd Annual Conference of the Cognitive Science Society* (p. 529). Austin, TX: Cognitive Science Society.
- Toyota, M., & Kitajima, M. (2010b). *The Organic Self-Consistent Field Theory*. Tokyo: ON-BOOK.ltd. (in Japanese)
- Toyota, M., Kitajima, M., & Shimada, H. (2008). Structured Meme Theory: How is informational inheritance maintained? In B. C. Love, K. McRae, & V. M. Sloutsky (Eds.), *Proceedings of the 30th Annual Conference of the Cognitive Science Society* (p. 2288). Austin, TX: Cognitive Science Society.

Modeling Behavior of Attention-Deficit-Disorder Patients in a N-Back Task

Jonathan Kottlors (jonathan.kottlors@mars.uni-freiburg.com)

Daniel Brand (brandd@informatik.uni-freiburg.de)

Marco Ragni (marco.ragni@cognition.uni-freiburg.de)

Human Reasoning Lab, Center for Cognitive Science, University of Freiburg

Friedrichstr. 50, D-79098 Freiburg, Germany

Abstract

Cognitive performance in memory tasks, as measured by the N-back task, shows large differences between attention deficiency disorder (ADD) patients and controls. Recent findings indicate fewer anatomical differences, which, in turn, makes a cognitive modeling of the inherent information processes possible. In this article, we investigate these aspects for 0-back and 1-back problems and introduce a new cognitive model explaining differences between patients and controls. The ACT-R model explains the results by differences in the utility and reward function. The presented methodology – using cognitive modeling for controls and patient groups – works toward two goals: (1) A potential explanation of a source of differences for a medically relevant group and (2) to present and evaluate a new approach using cognitive modeling in psychiatric research areas to identify more detailed descriptions of mental differences.

Keywords: N-Back, ACT-R, Attention Deficiency Disorder, Working Memory, Modeling of Disorders

Introduction

The so-called N-back task is used as a psychological tool to determine individuals' cognitive performance on (spatial) working memory problems. The participant is presented with a sequence a_1, \dots, a_m of letters one at a time. In a 1-back task, the participant is required to press a button if the letter being presented (a_i) is identical to its direct predecessor (i.e., if $a_i = a_{i-1}$ for the 1-back task). An N-back problem requires a comparison of the current letter (a_i) with the letter presented N-steps back (a_{i-N}). To keep the predecessor and successor in memory can require some sort of spatial representation. From a psychological perspective, a successful solution requires concentration and the ability to keep track of a number of items in working memory. A central finding is that response latency increases and accuracy decreases with increasing memory load (Braver et al., 1997; Cohen et al., 1994; Lovett, Daily, & Reder, 2000). Performance errors can be classified as errors of omission and errors of commission. An error of omission is committed if the participant does not press the button although the letters are identical and an error of commission if the participant presses the button although the letters to be compared are different. Errors of omission rarely appear for smaller N-back tasks (Braver et al., 1997), but are generally more likely than errors of commission.

Attention Deficiency Disorder (ADD) patients may perform more errors of commission as they are primarily characterized as having a much higher level of inattention, distractibility, and impulsiveness (Pary et al., 2002; Barkley, 1998). Although the exact biological or cognitive mechanisms are still unknown, a number of involved pathways have

been discussed (Pary et al., 2002). ADD can negatively affect the educational and social performance of those suffering from its symptoms (Pary et al., 2002) and it is one of the most common mental disorders, according to data from the NIH (NHS, 2008). Although there are differences in brain development, there are no known anatomical differences. An interesting insight we can draw from these findings is that attentional problems may depend on internal information processing rather than on physiological aspects of the brain. With this in mind, it seems possible to model this effect with a symbolic cognitive model. In order to test this hypothesis we decided to model the results of a study conducted by Klein, Wendling, Huettner, Ruder, and Peper (2006). The methodological approach of comparing clinical abnormalities with controls on a level of abstraction to identify cognitive distinctions within a cognitive model is, as described in this paper, a fairly new approach. For this reason, the general procedure in this study can serve as an example for other fields of clinical diagnostics especially for cognitive disorders. Similar approaches with different starting point question could be found in Hussain and Wood (2009).

State-of-the-Art

The Experiment by Klein et al. (2006)

We briefly report the empirical findings from Klein et al. (2006). They investigated different cognitive parameters of intra-individual variability to identify subgroups of ADD-patients in comparison with controls. All members of the patient-group (57 subjects) were patients from "Caritashaus Feldberg" a clinic specializing in the treatment of ADD diagnosis and met the criteria according to ICD-10 at the time of the study. They were diagnosed by experienced clinical psychologists and psychiatrists on the basis of Conners' parent and teacher rating scales (Steinhausen, 2000). The patient group included 49 boys (85.9 %) and 8 girls (14.1 %) with a mean age of 126.4 ± 21.2 months (range: 84-169 months) and a mean IQ of 96.6 ± 13 . The control group was matched to the patient-group. For this reason the controls do not differ in the mean age (126.9 months ± 21.7) or gender distribution (8 girls, 45 boys). Only the mean IQ (110.2 ± 12.82) was significantly ($t_{108} = 5.42, p = .001$) higher than in the patient-group. Further group descriptions are given in Klein and colleagues (2006).

Design, Method & Procedure. Both groups were tested with 0-back, 1-back, and 2-back problems. In each condition

(0-back, 1-back, and 2-back tasks) the participants were presented with exactly 100 trials. Each stimulus was presented for 0.5 seconds and the next stimulus was presented 2.0 seconds later. The “event” condition in the 0-back condition was the presentation of the letter “E” while all other letters were characterized as “nonevents.” In the 1-back condition, an event occurred if the letter being presented (a_i) was identical to the letter presented one step earlier (a_{i-1}), as characterized above. An event occurred randomly in about 20% of the trials. Ten practice trials preceded the 0-/1-back tasks; 20 the 2-back tasks. Subjects had to press the right-hand response button as quickly as possible for events, and the left-hand response button for nonevents.

Results. The authors decided to eliminate the 2-back condition as 17 patients and 5 controls had difficulties understanding the task. Therefore, they only report the results for 0-back and 1-back tasks. Three of the cases showed significant group differences, where the differences stand out in the 1-back condition. All results can be found in Figure 1. For further information please refer to Klein et al. (2006).

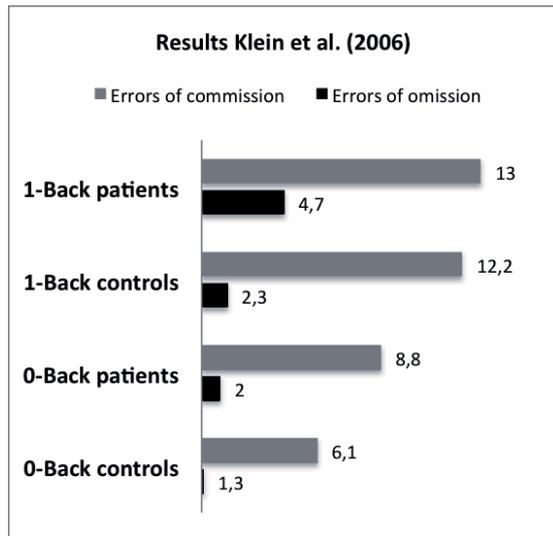


Figure 1: The empirical results for both ADD-patients and controls from the study of Klein and colleagues (2006).

ACT-R. The cognitive architecture ACT-R 6.0 (J. R. Anderson et al., 2004; J. Anderson, 2007) provides a number of working memory specific modules which are in turn associated with specific cortical regions. The smallest information unit is called a chunk. Chunks are modified by mental operations. These mental operations are associated with so-called production rules, which consist of a condition and an action part. The procedural module controls ACT-R’s strictly serial behavior; only one production at a time can fire. The likelihood that a production rule can fire depends on several

factors: First of all, the condition of the rule must be satisfied. If several production rules compete then noise and the production rules’ utility performs the selection process.

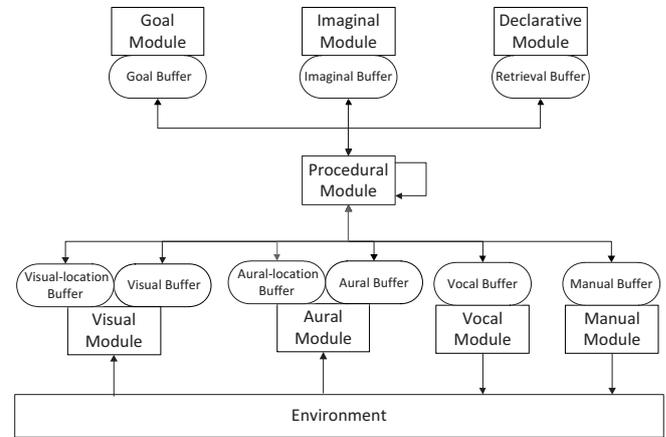


Figure 2: Overview of the different buffers and modules in the cognitive architecture ACT-R 6.0. The symbolic structure is represented by productions while subsymbolic parts control the execution of productions (based on utility) and the velocity of information retrieval from declarative memory and are therefore especially responsible for learning.

There have been several attempts to model the N-back task in ACT-R. One of the first cognitive models for the N-back was presented by Lovett et al. (2000). They identified two qualitatively different strategies used by participants: The so-called activation strategy, where participants respond ‘match’ if a letter seems familiar. A second strategy – the update strategy – involves actively maintaining a list of prior letters and updating that list after each letter is presented. They concluded that, working memory resources are not involved in the first strategy as no maintenance is involved, but are involved in the second. The ACT-R task model for the N-back task includes all the declarative and procedural knowledge necessary for performing this task according to the update strategy. In this model, each stored letter was represented as a declarative chunk indexed according to how many letters back it was from the current letter. They used mostly default values (see Anderson and Lebiere (1998)) and set activation noise to 0.04 and the retrieval threshold to 1.80 to optimize the fit to the data. Individual differences between participants could be captured by the previously attained source activation, parameter W, a type of attentional activation that is divided equally among the items in the current focus of attention. It spreads from these items to related chunks.

Juvina and Taatgen (2007) empirically investigated N-back tasks where subjects received feedback for each action. They successfully modeled the influence of feedback on learning rates. They found a significant change in the relation between omissions and commission error rates – as in Klein and colleagues (2006).

Aim of this research approach

The aim of this research approach was to identify relevant parameters with an influence on the deficit in cognitive performance and to provide a framework for further research approaches. Relevant symbolic processes should be identified with different kinds of modeling. The database from Klein et al. (2006), that examined ADD-patients and controls, served for the validation of the different models of the N-back task.

In addition, we briefly review the existing cognitive models for the N-back task. We discuss our findings in comparison to the experiment by Klein et al. (2006) and draw conclusions for further medical methods.

Methodology

In the following our research and methodological approach are introduced (see Figure 3). Note that the following elaborations of the cognitive modeling are all based on the empirical findings by Klein (2006) described in the “state of the art” part in this paper.

The Cognitive Model

One of the main conclusions we draw from the previous experiment and the literature is that ADD-patients may have a higher iniquity (see above and Pary et al., 2006). This *iniquity* shows itself in the behavior of the participants: in a heightened probability of pressing a button (to commit an error of commission) and a slightly greater level of difficulty in retrieving information from declarative memory. This is due to a greater amount of noise and results in errors of omission. The noise levels in our two patients models vary only slightly (.35 vs .362) in comparison to the control group model. Higher iniquity has not yet been modeled (e.g., Gunzelmann, Moore, Gluck, Van Dongen, & Dinges, 2009). We decided to represent the increasing iniquity by a higher negative utility or differences in the reward parameter. Our models do not assume other differences. As shown in Figure 6, the two patient-models (reward/utility) both lead to the results found by Klein (2006). Therefore these two are different forms of the original model (control group) with differences in only one specific parameter. These parameter differences are shown in Figure 4 and Figure 5.

Stimulus presentation times were set – as in Klein (2006) – to 0.5 seconds with 2 second breaks between the stimuli. The probability for N-back occurrence was set to 20%. Each participant evaluated a trial of 100 stimuli.

The 0-back task

In the 0-back task participants were required to find a specific letter in the given series of letters.

The variability of results is caused by specific parameter settings (as indicated in Table 1). Overall, two settings were considered as a plausible explanation for the error structure of the patients group: a “reward-model” and a “utility-model.” In these two models, either the reward parameter or the parameter of utility was varied (see Figure 4). Both models can explain differences in performance between patients and

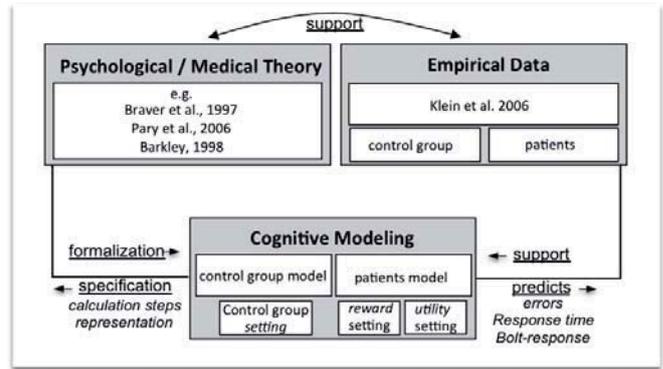


Figure 3: Research approach in this study. Empirical data for control group and patients were taken from Klein et al. (2006). Our modeling is orientated on these results. In the “modeling part” we introduce two different models for the patients group to predict the ADD group results.

controls (see Figure 3). From a psychological perspective arguments for both models are possible. On the one hand, the “utility-model” justifies the higher number of errors of commission by patients through a more intense use of the “N-back-found” production than in the control group model. On the other hand, the “reward-model” explains this particular error pattern with different reward setting. On a psychological basis this would mean the use of the “N-back-not-found” production has a more negative reward. Each “no-event-sequences” will decrease the use of the “N-back-not-found” production and this results in a higher amount of commission errors.

In this sense patients and controls differ in the following sub-symbolic parameters (see Table 1 and Table 2). As already mentioned in the “Cognitive Model” section you can see the different parameter settings in the differences between the two patient-models in Figure 4. For example a $\Delta R: 0$ in the parameter reward ($\Delta R: 0$) means that the fitting in this parameter for that model is not different from the two patient models or the control group model. A $\Delta U: 1.95$ in the “N-back-found” production means that the patient utility model has an 1.95 higher utility parameter than the control group model and also than the reward model, because their utility was not modeled.

Table 1: Parameters used to model patients and controls for the 0- & 1-back task.

	egs	:esc	:rt	:lf	:ans	:bll
Controls	1	t	-.5	0.25	0.35	0.5
Patients	1	t	-.5	0.25	0.362	0.5

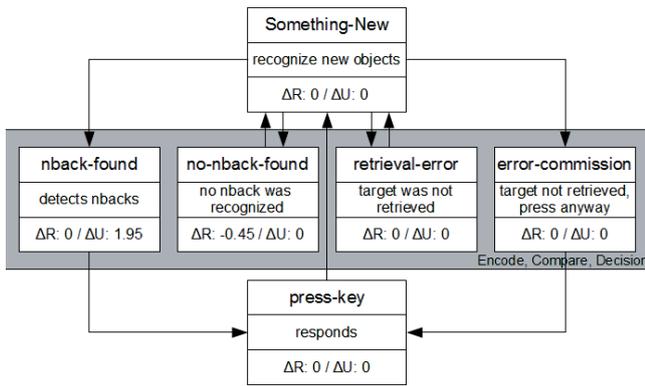


Figure 4: Diagram of the 0-back model “Something-new” and “press-key” productions and the encode-, compare- and decision-layer with its productions in the center. ΔR describes the difference between the patient and control group models in the parameter “reward” ($R(\text{patients}) - R(\text{control group}) = \Delta R$) for each specific production. This situation also applies to the alternatively designed utility model (ΔU) of the 0-back task.

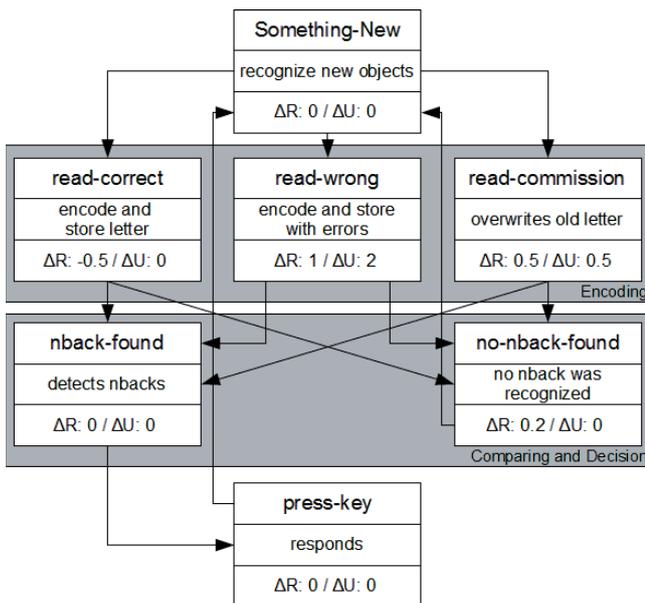


Figure 5: Diagram of the 1-back model. “Something-new” and “press-key” productions and the encode-, compare- and decision-layer with its productions in the center. ΔR describes the difference between the patient and control group models in the parameter “reward” ($R(\text{patients}) - R(\text{control group}) = \Delta R$) for each specific production. This situation also applies to the alternatively designed utility model (ΔU) of the 1-back task.

The 1-back task

In the 1-back task participants have to press a button, if the letter being presented is identical to the previous letter. This

Table 2: The base level activations for each stimulus in the 0-back task. Target stimulus for the remaining letters.

	target stimulus	remaining letters
Baselevel activation	50	0.5

task differs to the previous one so we included more production rules – because it cannot be reduced to a simple “retrieve-model” like the 0-back-task.

For this reason we introduced a more complex representation. Thus, the encoding layer is now separated from the comparing and decision layer and is also a possible source of errors.

As described in the explanation of the 0-back-task, we created two models that explain the differences between patients and controls. One model only uses the differences in utility and one uses the reward parameter (see Figure 5).

Results and Discussion

The present modeling replicates data from Klein (2006) satisfactorily (see Figure 6). Although both models match the original data in a sufficient way, the “reward-model” provides the better data. On the one hand, the scatter of the data in the “reward-model” is less around the mean of the behavioral data, and also comes closer to this. Also working memory components in the “compare and decision” layer in the 0- and 1-back-models like the “nback-found” production do not differ in the two groups looking at the “reward-model” (see Figure 6). This showed that there are no parameter differences in this “execution parts” of the “reward-model”, but therefor in the reward of inhibitory parts like the “no-nback-found” production, which results in this specific arrangement of errors. This so created iniquity or impatience is the main cause of the specific error pattern. Impatience is also mentioned in relation with a phenomenon called “delay aversion.” Delay aversion is a behavior that was investigated in connection with ADD and ADHD (Sonuga-Barke, Taylor, Sembi, & Smith, 1992). Our results indicate that the specific error pattern that ADD patients produce in an N-back task could be traced back to variations in the reward parameter of production rules. This could mean that even at this subconscious level “reward similar actions” are responsible for the underlying pathological behavior. The fact that this small variations in the reward parameters, could affect working memory capacity in a cognitive task was shown by the “reward model”. A selective control of the occurrence time concerning the N-back targets could clarify this question more. Longer “no-event phases” in the N-back task should lead to larger number of errors in the patient-groups than in control groups.

General Discussion and Outlook

There are only a few approaches to modeling different patient groups in ACT-R. Thus far, there are – to the best of our knowledge – no (symbolic) cognitive models of working

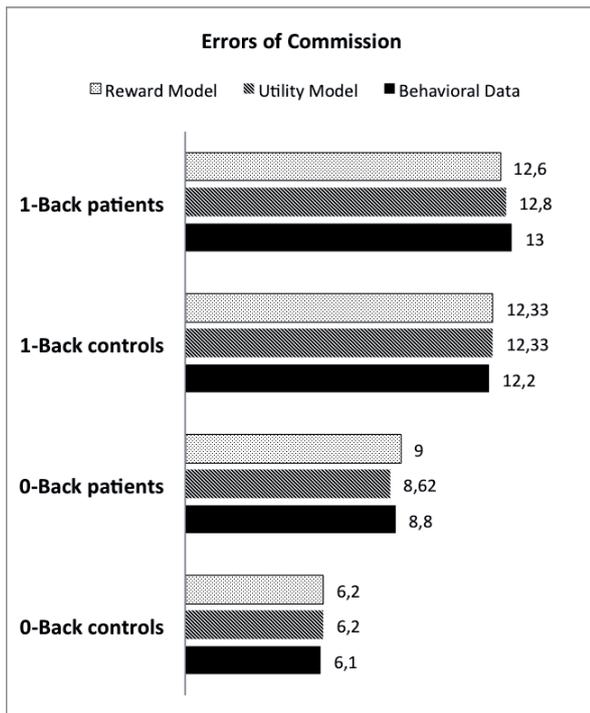
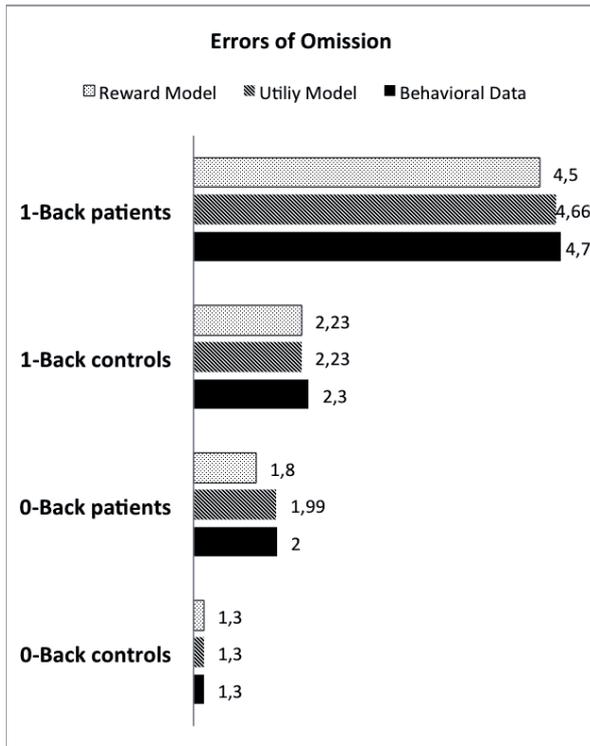


Figure 6: The identified errors of omission- and commission both for ADD-Patients and the predicted ACT-R data for the 0-back and 1-back task.

memory differences in ADD-Patients. One reason might be the difficulty of assessing differences in the attention span. Most cognitive models have assessed this purely by differences in parameter settings. ADD-patients represent an interesting and increasingly important population. By analyzing empirical data we decided to model increasing iniquity (resulting in a heightened probability of pressing a button (to commit an error of commission) and a slightly greater level of difficulty in retrieving information from declarative memory) by increasing negative utility. In the modeling process, it became apparent that there is at least an additional explanation pattern, namely, in differences with reward (keeping utility constant). A second “reward”-model was able to capture the empirical differences. Both models make a satisfactory prediction of the results and not only in parametric distances. This goes along with analysis from (Roberts & Pashler, 2000). He argued that parameters might give a good hint about a cognitive model if they constrain the outcomes. In this way, we have differentiated between two likely model. But, both models differ regarding an important aspect, namely the length of the presented sequence. The higher the length of the total sequence of presented letters, the higher the errors the reward-model would predict but not the utility model. A further prediction would be: Most psychological studies have presented participants with a random function based on the probabilities with which an event occurs. But, based on the cognitive modeling, we would predict that higher pauses between events might trigger more commission errors in the patient groups than in the controls. These questions have not yet been investigated empirically or reported in the literature. This shows the power of the cognitive model approach’s ability to make clear predictions based on an algorithmic implementation, which can, in turn, be empirically investigated in human experiments and suffice to discern different theories.

Our modeling approach is purely information-theoretic, i.e., it does not require anatomical differences nor does it explain the effects solely by parameter settings. Recently, Gunzelmann and colleagues (2009) modeled the decline of cognitive ability by sleep-deprived soldiers in an attention task. They applied a Psychomotor Vigilance Test (Dinges & Powell, 1985; Dorrian, Rogers, & Dinges, 2005). The task of the participants was to monitor a known location on a computer screen and to press a response button each time a stimulus appeared at random intervals between 2 seconds and 10 seconds. Sleep deprivation had a severe effect on performance due to decreased alertness. Differences between groups were explained by differences in parameters. Certainly the parameter fitting process provides no explicit information about prevalent biological differences, but several findings can still be derived in this sort of modeling.

Unsurprisingly, results showed that a higher level of complexity in the N-back task leads to a change in the relationship between omission- and commission errors. Our hypothesis in this context argues that at whenever a complexity-related

omission error is committed, a commission error is counted. This only applies to 2-Back and higher N-backs because of a probable miscounting. This has not been reported so far, but might indicate that this aspect has led to a distortion of reported empirical findings in the literature. These findings indicate that there is a great importance to develop a trustworthy method to identify the reasons for an error to occur. Effects of training of working memory by a dual-version N-back tasks on intelligence has been investigated (Jaeggi, Buschkuhl, Jonides, & Perrig, 2008). Nab et al. (2009) reported changes on a neuronal level using the N-back task also for cognitive training. These two findings reveal further investigations on the N-back task to develop new therapeutic methods for ADD. Future work will investigate the differences regarding neurological predictions (fMRI-Analysis) and the modeling of subgroups of ADD-Patients. Cognitive disorders might be – at least in a non-negligible part – traced back to differences in mental model operations, which are linked to the production buffer that has been associated with basal ganglia. Further psychological and medical research (to explain the underlying medical condition) is necessary.

Acknowledgments

The authors are grateful to Christoph Klein (University of Bangore, UK) for various helpful discussions. This research was partially supported by the DFG (German National Research Foundation) in the Transregional Collaborative Research Center, SFB/TR 8 within project R8-[CSPACE].

References

- Anderson, & Lebiere. (1998). *The atomic components of thought*. Mahwah: Routledge.
- Anderson, J. (2007). *How can the human mind occur in the physical universe?* Oxford: Oxford University Press.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, 111(4), 1036-1060.
- Barkley, R. A. (1998). *Attention deficit hyperactivity disorder: A handbook for diagnosis and treatment* (2nd ed.). New York: Guilford Press.
- Braver, T. S., Cohen, J. D., Nystrom, L. E., Jonides, J., Smith, E. E., & Noll, D. C. (1997). A parametric study of prefrontal cortex involvement in human working memory. *NeuroImage*, 5, 49-62.
- Cohen, J. D., Forman, S. D., Braver, T. S., Casey, B. J., Servant-Schreiber, D., & Noll, D. C. (1994). Activation of prefrontal cortex in children during a nonspatial working memory task with functional mri. *Human Brain Mapping*, 1, 293-304.
- Dinges, D. F., & Powell, J. W. (1985). Microcomputer analyses of performance on a portable, simple visual rt task during sustained operations. *Behavior Research Methods, Instruments, & Computers*, 17(6), 652-655.
- Dorrian, J., Rogers, N. L., & Dinges, D. (2005). Psychomotor vigilance performance: Neurocognitive assay sensitive to sleep loss. In C. A. Kushida (Ed.), *Sleep deprivation. clinical issues, pharmacology, and sleep loss effects* (p. 39-70). New York, NY: Marcel Dekker.
- Gunzelmann, G., Moore, L. R., Gluck, K. A., Van Dongen, H. P. A., & Dinges, D. F. (2009). Examining sources of individual variation in sustained attention. In N. Taatgen & H. van Rijn (Eds.), *Proceedings of the thirty-first annual meeting of the cognitive science society* (p. 608-613). Austin, TX: Cognitive Science Society.
- Hussain, F., & Wood, S. (2009). Computational modeling of deficits in attentional networks in mild traumatic brain injury: An application in neuropsychology. In *Proceedings of 31th annual conference of the cognitive science society* (p. 2675-2680). Amsterdam.
- Jaeggi, S. M., Buschkuhl, M., Jonides, J., & Perrig, W. J. (2008). From the cover: Improving fluid intelligence with training on working memory. *Proceedings of the National Academy of Sciences*, 105, 6829-6833.
- Juvina, I., & Taatgen, N. (2007). Modeling control strategies in the n-back task. In *Proceedings of the 8th international conference on cognitive modeling*. Ann Arbor, Michigan, USA.
- Klein, C., Wendling, K., Huettner, P., Ruder, H., & Peper, M. (2006). Intra-subject variability in attention-deficit hyperactivity disorder. *Biological Psychiatry*, 60(10), 1088-1097.
- Lovett, M. C., Daily, L. Z., & Reder, L. M. (2000). A source activation theory of working memory: cross-task prediction of performance in act-r. *Cognitive Systems Research*, 1(2), 99-111.
- Nab, F., Varrone, A., Farde, L., Jucaite, A., Bystritsky, P., Forssberg, H., et al. (2009). Changes in cortical dopamine d1 receptor binding associated with cognitive training. *Science*, 323, 800-802.
- NHS. (2008). *Cg72 attention deficit hyperactivity disorder (adhd): full guideline*. PDF. (retrieved 2008-10-08)
- Pary, R., Lewis, S., Matuschka, P. R., Rudzinskiy, P., Safi, M., & Lippmann, S. (2002). Attention deficit disorder in adults. In *Annals of clinical psychiatry* (Vol. 14, p. 105-111). Springer.
- Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? a comment on theory testing. *Psychological Review*, 107, 358-367.
- Sonuga-Barke, E. J., Taylor, E., Sembi, S., & Smith, J. (1992). Hyperactivity and delay aversion–i. the effect of delay on choice. *Journal of Child Psychology and Psychiatry, and Allied Disciplines*, 33(2), 387-398.
- Steinhausen, H. (2000). *Psychische Störungen bei Kindern und Jugendlichen*. München: Urban and Fischer.

Combining cognitive modeling and EEG to predict user behavior in a search task

Felix Putze (felix.putze@kit.edu), Johannes Meyer, Joscha Borné, Tanja Schultz

Karlsruhe Institute of Technology (KIT), Germany

Daniel V. Holt (daniel.holt@psychologie.uni-heidelberg.de), Joachim Funke

University of Heidelberg, Germany

Keywords: user modeling; HCI; EEG; spectral analysis; search task; cognitive modeling; temporal difference learning

In recent years, the development of intuitive human-computer interfaces has drawn a growing interest from different research communities. One major aspect of designing such interfaces is to measure or predict the cognitive state and cognitive processes of the user. This ability would enable systems to optimize information presentation for the available cognitive resources, current goals, and cognitive context of the user and give support in critical or difficult situations.

Cognitive models have in the past successfully been applied to explain and predict user behavior and build interactive systems that respond adaptively to user state (e.g., Peebles & Cox, 2006; Ritter, Anderson, Koedinger, & Corbett, 2007). More recently, cognitive models have been used to predict neural activity in fMRI studies or to infer cognitive states in combination with fMRI data (e.g., Anderson, Albert & Fincham, 2005; Anderson, Betts, Ferris & Fincham, 2010). This approach is particularly interesting from an HCI perspective, as neural data may serve as indicators of cognitive processes that have no easily observable behavioral correlates (e.g., decision making or memory retrieval). In the present study we bring together these strands of research by combining cognitive models with neural data derived from EEG recordings to predict user behavior. To evaluate this approach, we used a paradigm based on a memory-guided search task previously employed by Fu and Anderson (2006). We chose this type of task as it resembles a basic type of human-computer interaction, has a structure suitable for an EEG study, and has successfully been modeled by applying a form of temporal difference learning (cf. Sutton & Barto, 1998) implemented in the ACT-R cognitive architecture.

The focus of our analysis was (1) to explore which electrophysiological measures are suitable indicators for various aspects of cognitive user state in this task and (2) to investigate how the behavioral predictions of the cognitive model of the task could be enhanced by integrating the neural data.

Method

The task was adapted from the maze search task used by Fu & Anderson (2006), which is a multi-step decision and learning paradigm. In this task, participants have to find a goal node in a hierarchical tree structure. For purposes of this study, the tree structure was presented as a file system in which a particular file had to be located. Participants start each trial at the root node of the tree. At each node, they select one of two options leading to the next node until they reach a leaf node. When the leaf node contains the target, the next trial starts. When reaching a dead end the participant is reset to the node with the last correct decision. The task consists in discovering and consistently applying the correct route from the root node to the target. To allow the manipulation of associative learning requirements, each node is labeled with a word randomly selected from a set of concrete nouns unique to the node (representation set). The correct decision option for each node is tied to this label and has to be learned. The easy version of the task used a representation set of two concrete nouns per node, the difficult version a set of four. Decision options were abstractly labeled “alpha”, “beta”, “gamma” and “delta” in order to eliminate spatial or semantic cues.

The task was displayed on a computer screen, showing the folder in the center and the two decision options to the left and right below the folder at each stage of the task. Responses were given by pressing one of four buttons. After instruction and practice each participant completed the task for two easy search trees for 60 trials and two difficult search trees for 120 trials. During task performance EEG was recorded from 32 channels using actiCAP active electrodes and an actiCHamp amplifier (BrainProducts, Germany). In a completed pilot study, we obtained data from 18 participants (age 19 to 26, five female).

Data Analysis

Detailed data analysis is currently in progress and will be reported in full at the conference presentation. Here, we present a short summary of the cognitive model, selected preliminary results for the behavioral data, and one potential electrophysiological indicator of learning performance.

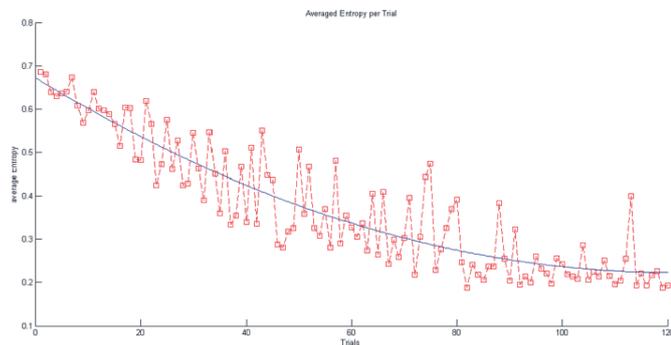


Figure 1: Development of entropy of decision probabilities for one participant in a difficult search tree; displayed per trial and as running average.

Following Fu & Anderson (2006), we constructed a computational model based on temporal difference learning (TD) to simulate individual learning progress and predict the certainty of an upcoming decision. To model individual learning progress, the TD model is trained with the history of participants' decisions for each trial block in order to predict the next decision. We were able to reproduce the basic pattern of results reported in Anderson and Fu: The model correctly predicts faster convergence for nodes closer to the leaves of the tree as reward is only received there and propagated backwards. Additionally, we used a Softmax function to derive decision probabilities from the learned scores and calculate the entropy of the resulting distribution to derive decision confidence (Figure 1). This information can be used to identify trials traversing states for which learning has not converged yet, even if no actual error occurred.

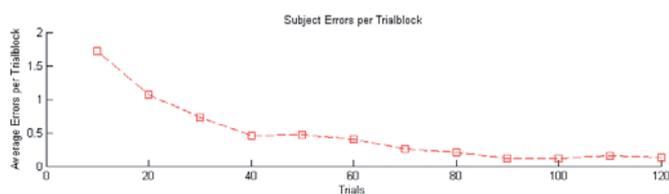


Figure 2: Error per trial block averaged over all participants for difficult search trees.

We expected to observe correlates of a number of different cognitive processes in the EEG signal. For example, we assumed that the power of oscillations in the theta band increases during memory activity (Onton, Delorme & Makeig, 2005; Klimesch, 1998). A qualitative comparison of the aggregated behavioral data and averaged theta power in a frontal region shows that this may indeed be the case: In Figure 2, the average number of errors per trial for difficult search trees is displayed. As expected, this value drops sharply during early trials, rises again between trials 40 and 60 as interference between learned items increases, before finally converging towards a lower limit. The intermediate increase in the number of errors seems to be mirrored by a

corresponding rise in theta power for those trials, as shown in Figure 3.

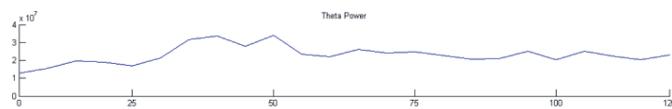


Figure 3: Spectral power in the theta band at F3 averaged over recordings for difficult search trees of three participants.

Conclusion and Future Work

The preliminary results suggest that the basic computational model is adequate and that there may be valid electrophysiological correlates of learning progress, which is a promising basis for the planned detailed analyses. Future work will concentrate on a formal integration of computational and EEG-based prediction in a Bayesian framework and the addition of further electrophysiological markers, e.g., related to decision making or error feedback. Additionally, the paradigm offers several possibilities for extension by further varying its cognitive demands (e.g., the size of the search tree or the representation set) or its semantic framing (e.g., as a web search or spatial navigation task).

References

- Anderson, J., Albert, M. & Fincham, J. (2005). Tracing Problem Solving in Real Time: fMRI Analysis of the Subject-paced Tower of Hanoi. *Journal of Cognitive Neuroscience*, 17, 1261-1274.
- Anderson, J. R., Betts, S., Ferris, J. L., & Fincham, J. M. (2010). Neural imaging to track mental states while using an intelligent tutoring system. *PNAS*, 107, 7018-7023.
- Fu, W. & Anderson, J. (2006). From Recurrent Choice to Skill Learning: A Reinforcement-Learning Model. *Journal of Experimental Psychology: General*, 135, 2, 183-206.
- Klimesch, W. (1998). EEG alpha and theta oscillations reflect cognitive and memory performance: a review and analysis. *Brain Research Reviews*, 29, 169-195.
- Onton, J., Delorme, A. & Makeig, S. (2005). Frontal midline EEG dynamics during working memory. *NeuroImage*, 27, 341-356.
- Peebles, D. J. & Cox, A. L. (2006). Modelling Interactive Behaviour with a Rational Cognitive Architecture. In P. Zaphiris & S. Kurniawan (Eds.). *Human Computer Interaction Research in Web Design and Evaluation*, London: Idea Group.
- Ritter, S., Anderson, J. R., Koedinger, K. R., & Corbett, A. (2007). Cognitive tutor: Applied research in mathematics education. *Psychonomic Bulletin & Review*, 14, 249-255.
- Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.

Towards a 50 msec Cognitive Cycle in a Graphical Architecture

Paul S. Rosenbloom (Rosenbloom@USC.Edu)

Department of Computer Science and Institute for Creative Technologies
12015 Waterfront Dr., Playa Vista, CA 90094 USA

Abstract

Achieving a 50 msec cognitive cycle in any sufficiently sophisticated cognitive architecture can be a significant challenge. Here an investigation is begun into how to do this within a recently developed graphical architecture that is based on factor graphs (with the summary product algorithm) and piecewise continuous functions. Results are presented from three optimizations that leverage the structure of factor graphs to reduce the number of message cycles required per cognitive cycle.

Keywords: Cognitive architecture, cognitive cycle time, graphical models, optimization.

A common assumption underlying many cognitive architectures is that there is a core cognitive cycle that runs at ~50 msec/cycle, the time scale of the quickest human responses (once peripheral processing, such as physical movement, is subtracted out). This value is close to the 70 msec mean originally given for the Model Human Processor (Card, Moran & Newell, 1983), and matches the value used in ACT-R (Anderson, 2007), EPIC (Kieras & Meyer, 1997) and Soar (Laird, 2012). Hitting such a rate in reality is critical for architectures that are to model cognition in real time as well as for architectures that are to support construction of intelligent systems that operate on human time scales. It is less critical when the focus is purely on the non-real-time modeling of human cognition; but even there it matters in principle whether the approach can reach this time scale within neurobiological implementation constraints, as well as in practice whether the model can run fast enough for serious experimentation on complex tasks.

Driven by this constraint, there has been considerable recent work on improving the efficiency and scaling of architectural capabilities such as declarative (semantic) memory (Derbinsky, Laird & Smith, 2010; Douglass & Myers, 2010), as well as a longer history of such efforts that go back at least to work on the efficiency and scaling of procedural (rule) memory (Forgy, 1982; Doorenbos, 1993). The focus in this article is on improving the efficiency and scaling of a form of graphical model (Koller & Friedman, 2009) that is being explored as an implementation level for a broad spectrum, tightly integrated and functionally elegant *graphical cognitive architecture* (Rosenbloom, 2011a&b).

Graphical models were chosen as the basis for this architecture because of their potential for yielding a uniform approach to implementing and integrating together state-of-the-art algorithms across symbol, probability and signal processing. At their core, graphical models provide efficient computation over complex multivariate functions

by decomposing them into the product of simpler subfunctions and then mapping the results onto networks of nodes and links. In a factor graph – the most general form of graphical model and the one used in the architecture – *variable nodes* represent function variables, *factor nodes* represent subfunctions, and *links* connect subfunctions with their variables (Kschischang, Frey & Loeliger, 2001). Multiple inference algorithms exist for such graphs, both exact and approximate. The graphical architecture uses a variant of the summary product algorithm (Kschischang *et al.*, 2001), a message-passing scheme that is exact for non-loopy graphs and approximate for loopy ones.

Given this algorithm, a single cognitive cycle maps onto the architecture as a *graph cycle* (GC); a solution to the graph, given evidence concerning the values of some variables, generated by passing messages until quiescence and then updating working memory (Rosenbloom, 2011c). The graph roughly corresponds to long-term memory and the evidence to working memory. A single graph cycle can include parallel waves of rule firings, access to declarative knowledge, perception, and simple forms of reasoning (including fixed chains of probabilistic reasoning, as are found for example in POMDPs). Each graph cycle is itself composed of a sequence of *message cycles* (MCs), during each of which a single message is passed along one link.

Given this core capability, the graphical architecture has already been shown to support procedural and declarative memories (Rosenbloom, 2010), plus forms of perception (Chen *et al.*, 2011), imagery (Rosenbloom, 2011d) and problem solving (Chen *et al.*, 2011; Rosenbloom, 2011c). However, it still operates at a time scale that is too often far above the critical 50 msec/GC threshold. Prior to the work described in this article, the average time per GC – in LispWorks 6.0.1 on a 3.4 GHz Intel Core i7 iMac with 8GB of 1333 MHz DDR3 RAM – was close to the desired value for simple tasks, such as 55 msec for the one GC involved in accessing a small semantic memory. However, the Eight Puzzle averaged 872 msec/GC when run to completion on a problem that needed 9 GCs; and a more complex virtual navigation task (Chen *et al.*, 2011) – which combined perception (via a three-stage CRF), localization (via part of SLAM), and decision-theoretic choice (via a three-stage POMDP) – was even more problematic. Although this last task wasn't implemented until after some of the new optimizations described in this article were already in place, it still required 2288 msec/GC when run for 20 GCs, a factor of 46 too slow.

The obvious strategy for reducing these numbers is to decompose the problem into (1) reducing the number of

message cycles per graph cycle (MC/GC), and (2) reducing the time per message cycle (msec/MC); and then to tackle both of these subproblems individually. Across the three tasks just mentioned, the range of 55-2288 msec/GC decomposes into 564-3635 MC/GC and .1-.6 msec/MC. This article focuses on the first subproblem, exploring how to leverage the structure of the architecture’s factor graphs – and the dependencies that these implicitly define – to dramatically reduce MC/GC. Work on the second subproblem – which is exploring new representations for the functions and messages at the heart of the architecture (as proposed in Rosenbloom, 2011b) – is not as far along, and is thus left as future work. We begin here with additional relevant background on the architecture’s use of factor graphs and summary product, and on a set of early optimizations that were implemented prior to this work, before examining three new MC/GC optimizations.

Factor Graphs and Summary Product

In its simplest form, a factor graph embodies a variable node for each variable in the function of interest, a factor node for each subfunction in the product decomposition, and bidirectional links that connect each factor node with the variables it uses. Figure 1, for example, shows a factor graph for a polynomial function of three variables, with three variable nodes and two factor nodes. In more complex graphs, variable nodes may represent combinations of function variables, as in Figure 2, to exploit composite variables in the graph that are cross products of the function variables involved. Maintaining such cross products is crucial, for example, to solving the *binding confusion problem* (Tambe & Rosenbloom, 1994) by tracking which values of one variable are consistent with which values of another variable (Rosenbloom, 2011a).

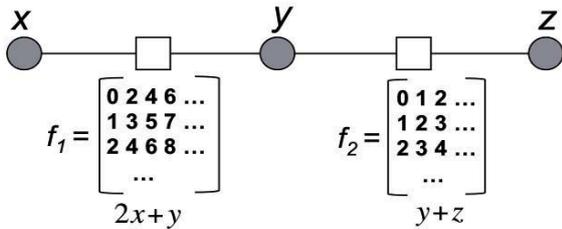


Figure 1: Factor graph for $f(x,y,z) = y^2 + yz + 2yx + 2xz = (2x+y)(y+z) = f_1(x,y)f_2(y,z)$

By definition, factor graphs are bidirectional, so wherever there is a link between two nodes, messages pass in both directions along the link. However, in creating the graphical architecture it became clear that introducing a form of unidirectional link would enable subgraphs corresponding to the kinds of conditions and actions that occur in standard rule-based procedural memories (as in Figure 2). Conditions match to information in working memory, combining their results so that actions can then propose changes to working memory. Declarative knowledge is encoded in terms of *condacts* – which combine the effects

of *conditions* and *actions* to pass messages both to and from working memory – plus functions, such as those associated with the two factor nodes in Figure 1. Condacts yield standard bidirectional subgraphs, while conditions and actions yield subgraphs with a single active direction for message passing. This notion of link directionality is not the same as that found in Bayesian networks; the former concerns the direction of message passing, while the latter concerns how variables functionally depend on each other in factor nodes (such as in defining conditional probabilities).

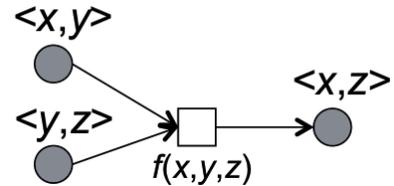


Figure 2: Factor graph for rule $\text{After}(x,y) \wedge \text{After}(y,z) \Rightarrow \text{After}(x,z)$.

Conditions, actions and condacts define variabilized patterns that are combined, along with functions, into *conditionals*, a generalized form of rule that forms the basic long-term-memory element in the graphical architecture. Figure 2, for example, shows a rule-like conditional for transitivity that is composed of two conditions and one action, and for which the links only pass messages towards the right. A bit of declarative memory might instead use two condacts and a function to specify the conditional probability of the first given the second, such as $\text{Concept}(x), \text{Color}(y): [(\text{Table: Brown}=.95, \text{Silver}=.05) (\text{Dog: Brown}=.7, \text{White}=.25, \text{Silver}=.05) \dots]$ for the classification of an object given its color. Table 1 lists basic statistics on the conditionals and their patterns for the three tasks that have been introduced. It can be seen that the Eight Puzzle is largely procedural, navigation is largely declarative, and semantic memory is more of a blend. This not surprisingly leads to a skewed distribution of link directions for the latter two, and a more balanced distribution for the first (Table 2).

Table 1: Conditional and pattern statistics for the Semantic Memory, Eight Puzzle and Navigation tasks.

	Conditionals	Conditions	Condacts	Actions
S	9	12	11	3
E	19	62	0	32
N	25	4	47	1

Table 2: Graph statistics for the Semantic Memory, Eight Puzzle and Navigation tasks.

	Nodes		Links	
	Factor	Variable	Uni-	Bi-
S	83	82	114	55
E	341	402	824	1
N	161	132	75	214

Each message along a link specifies a function over the variables in the link's variable node that constrains the variables' values. These functions are represented in the graphical architecture as piecewise continuous; in particular as doubly linked arrays of nD rectilinear (i.e., *orthotopic*) regions, where each variable maps onto a dimension, and the value function for each region is linear over its variables, as in Figure 3 (Rosenbloom, 2011b). If the function is Boolean, regions with a value of 1 are valid while regions with a value of 0 are not. If it is probabilistic, the function specifies the density over that region of variable values. However, functions can also mix these two, as in Figure 3, as well as approximate arbitrary continuous functions.

Given a new input message at a variable node, new output messages are computed for each of its links via a *pointwise product* of the new message with the incoming messages along all of the other links (except for the one along the output link). A pointwise product is like an inner product, where the value at corresponding points is multiplied; but there is no final summation over the result, so the output and input have the same rank. At a factor node, the input messages are likewise multiplied in this manner, but the factor function is also included in the product, and then all variables not in the output message are summarized out, by either integrating over them to yield marginals or maximizing over them to yield the MAP estimate. Figure 4 shows for example how evidence values of 3 for variable x and 2 for variable z propagate through the factor and variable nodes in the factor graph from Figure 1, to ultimately yield the marginal on variable y .

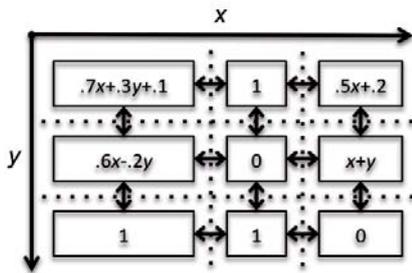


Figure 3: Piecewise continuous function as array of linear regions.

Given these early optimizations, MC/GC ranged from 564 for semantic memory to 1459 for the Eight Puzzle. With a slightly enhanced queuing scheme that will be described later, the navigation task required 3635 MC/GC. For comparison purposes, this enhanced form of queuing reduced the number of messages for semantic memory by 34% (to 371) and for the Eight Puzzle by 27% (to 1062). Without these early optimizations a usable system would have been infeasible from the start, and approaching 50

Both product and summarization are computed in the architecture by systematically stepping through the nD function(s) – following the links between adjacent regions – and at each step either multiplying the corresponding regions from two functions or summarizing out a dimension of a region within one function. Summarization involves either adding the integral of the region along the dimension to the current total or computing the maximum of the current region's max and the cumulative max so far.

At the beginning of each cognitive cycle, all messages are initialized before message passing begins. If a factor node has no inputs – as is true for working memory nodes (because changes to working memory occur at decision time rather than directly via message passing) and nodes that represent functions in conditionals (which actually appear in the architectural graph in a different manner than is shown in Figures 1 and 4) – the factor node's function (once unneeded variables are summarized out), becomes the initial outgoing message. Messages from all other nodes are initialized with a value of 1, yielding no initial constraint since such messages are identities for pointwise product.

All initial messages are placed into a global message queue, which is then continuously updated as existing messages are sequentially popped and processed, and new messages are generated. The cognitive/graph cycle reaches quiescence when there are no more messages in the queue.

Preexisting Optimizations

Several optimizations that reduce MC/GC were implemented early in the development of the graphical architecture. A form of dynamic programming was incorporated that caches and reuses the last message generated along each active direction of each link. In addition, to facilitate reaching quiescence with real functions, the cached message along each link direction was updated, and a new message added to the queue, only when the difference between the old and new messages exceeded $\epsilon = 10^{-7}$. To further reduce the number of messages to be processed, not all messages were inserted at the back of the queue. More constraining messages – ones that are 0 everywhere and thus halt all processing downstream from them, or ones that at least provide some information via values that vary over the variable's domain – were placed at the front of the queue, leaving only constant non-zero messages, which provide little discrimination, to be inserted at the back (see Figure 7a). The hope was for uninformative messages to be updated by new values along their link before being popped off the queue for processing.

Given these early optimizations, MC/GC ranged from 564 for semantic memory to 1459 for the Eight Puzzle. With a slightly enhanced queuing scheme that will be described later, the navigation task required 3635 MC/GC. For comparison purposes, this enhanced form of queuing reduced the number of messages for semantic memory by 34% (to 371) and for the Eight Puzzle by 27% (to 1062). Without these early optimizations a usable system would have been infeasible from the start, and approaching 50

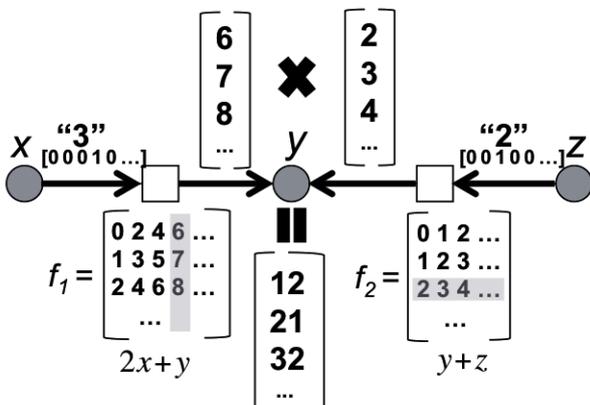


Figure 4: Computation via the summary product algorithm of the marginal on y from evidence on x and z .

msec/GC would have been impracticable. But, even with them, reaching this threshold still requires either: (1) reducing the worst-case MC/GC by 98% (from 3635 to 83), (2) reducing the worst-case msec/MC by 98% (from .6 to .014), or (3) some lesser combination of these reductions. The remainder of this article focuses on the first option.

Message Reuse Across Graphical Cycles

The early optimizations included caching of messages to enable their reuse across message cycles, but reinitialization of all messages was still required across graph cycles because there was otherwise no guarantee that modifying one message would result in all of the other messages in the graph being updated appropriately. Consider, for example, the loopy graph in Figure 5. If A is set to 0, D is set to 1, and there is no evidence concerning B and C, the graph converges to where all of the messages except the one from D are 0. If, on the next graph cycle, A becomes 1, all of the messages should settle to 1. However, without reinitialization the loop remains locked at 0. The new message to B, computed as the product of the new message from A (1) with the existing message from C (0), remains at 0, as does the new message to C. This contrasts sharply with, for example, the Rete algorithm for rule match, where messages (tokens) corresponding to unmodified regions of working memory can all be maintained and reused across cycles (Forgy, 1982).

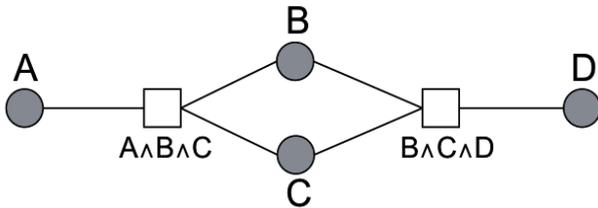


Figure 5: Loopy factor graph.

It does turn out, however, to be possible to identify segments within the overall factor graph where such reinitialization can be avoided, and where messages from the previous graph cycle can thus be reused. To do this requires preanalyzing the graph to determine which messages can possibly depend on factor node functions that may be modified between graph cycles; in particular, functions in working-memory factor nodes that are modifiable by decisions, and factor node functions specified in conditionals that are modifiable by learning. For each of these factor nodes a list of its *descendants* is first precomputed, where each descendant comprises: (1) a descendant node whose outgoing messages may be affected by messages originating at the modifiable node; and (2) a list of the descendant node's neighbors via which this influence may reach the descendant node. A message out of a node in the graph is then only reinitialized when: (1) the node is a descendent of a modifiable node that has actually been changed, and (2) the listed neighbors may pass this influence to the node so as to affect the output message.

Figure 6 shows a variant of the graph from Figure 5, but with some of the links now unidirectional, and both A and D modifiable (although shown as variable nodes, there would be a working-memory factor node feeding each). The descendants of A here are $(A_F; A_V)$, $(B; A_F, B^{\wedge}C^{\wedge}D)$, $(C; A_F)$, $(B^{\wedge}C^{\wedge}D; B, C)$, $(D; B^{\wedge}C^{\wedge}D)$. The descendants of D are $(B^{\wedge}C^{\wedge}D; D)$ and $(B; B^{\wedge}C^{\wedge}D)$. If the value of A (i.e., A_V) changes, all of the messages in the graph, except for the one from D, would need to be reinitialized; but if D changes, only two messages – from D to $B^{\wedge}C^{\wedge}D$ and from $B^{\wedge}C^{\wedge}D$ to B – would need reinitialization. All messages not reinitialized in this fashion are retained, allowing reuse of messages that are guaranteed to remain unchanged. This optimization cannot lower the number of message cycles during the first graph cycle, but it can in all later cycles.

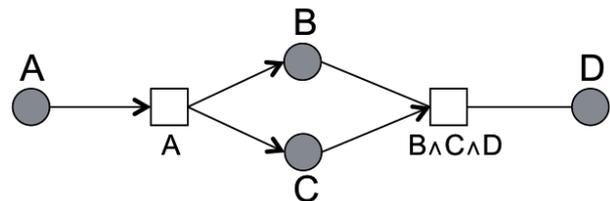


Figure 6: Variant of the loopy factor graph with a mix of bidirectional and unidirectional links.

The semantic memory test case is normally only run for a single graph cycle, but if a second GC is run without the evidence being changed, this optimization reduces the number of message cycles during the second graph cycle from 564 to 0 (saving 100%), yielding a total drop over the two graph cycles from 564 to 282 MC/GC (saving 50%). For the Eight Puzzle the number of message cycles during the second graph cycle drops from 1553 to 595 (saving 62%). Over the 9 GCs required to solve this particular problem, the average MC/GC dropped from 1459 to 1050 (saving 28%). With the navigation graph, the MC/GC drop (over 20 GCs) was from 3594 to 1359 (saving 62%).

Improved Message Ordering

The early optimizations included a heuristic for message insertion in the queue. The new approach to queuing retains the notion of constraint used there, while providing a more direct way of ensuring that messages that should be held until they contain appropriate content remain in the queue. Here we again preanalyze the graph structure, but this time to determine the *depth* of each link (in each direction). Links from nodes with no inputs – which again turn out to be working memory factor nodes plus factor nodes derived from functions in conditionals – have a depth of 0. For all other nodes not involved in loops – for which there is no unique depth – their depth is calculated as one plus the maximum of the depths of all neighbors from which they receive messages. The depth of a link in a particular direction is then simply the depth of its source node.

Message depth can then be used as a queuing heuristic that delays the processing of a message when there are

shallower ones – which thus could conceivably influence its content – also available in the queue. To implement this, the single original queue is split into a sequence of smaller queues. The first one is for *empty messages* (constant at 0) and the last one is for *full messages* (constant at 1). The former block all processing downstream from them, and can't be constrained any further. The latter are completely unconstrained, and thus not particularly useful. In between these two, a single queue was initially used for all other messages (Figure 7b), yielding the baseline results already presented for the navigation task. However, this has since been extended further, stratifying these other messages into a sequence of intermediate queues based on their depth.

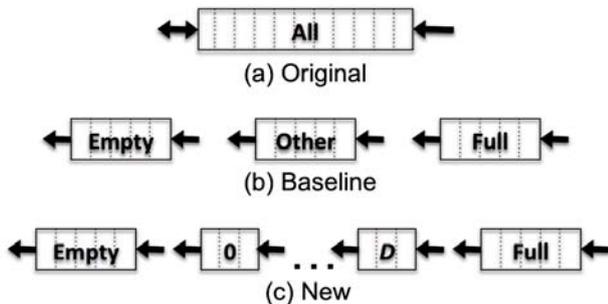


Figure 7: Three queue disciplines explored.

As shown in Figure 7c, one intermediate queue is created for each possible node depth – ordered from smallest to largest – for a total number equal to one plus the *depth of the graph*; i.e., the maximum of the depths of all of the nodes in the graph (D): 29 for semantic memory, 45 for the Eight Puzzle, and 76 for navigation. The last intermediate queue also handles links affected by loops. By stratifying messages in this manner, messages deeper in the graph that can be affected by shallower processing are delayed until all shallower messages are processed.

When all of the queues are included, empty messages are always sent before any other messages are considered. If there are no empty messages, then the intermediate queues are tried according to increasing depth. If there are no messages in any of these queues, the full-message queue is drained. When there are no messages in any of the queues, quiescence has been reached.

This optimization can help even during the first graph cycle, and can handle links along which messages are passed bidirectionally, as long as there are no loops. For messages affected by loops, ordering is essentially reduced to the previous baseline, with just one intermediate queue. This optimization, when enabled by itself, reduces MC/GC from the original single queue version by 60% (to 224) in semantic memory and by 43% (to 826) in the Eight Puzzle. In comparison to the three-queue baseline this is a savings of 40% for semantic memory and 22% for the Eight Puzzle. Improvement from this baseline in the navigation task lowers MC/GC by 86% (to 503).

When both this optimization and the previous one are combined, MC/GC drops by 61% (to 224) over one GC of semantic memory and by 90% (to 112) over two GCs. For

the Eight Puzzle, MC/GC drops by 59% (to 602) over the 9 GCs. The gains from the three-queue baseline are 40% over one GC of semantic memory and 70% over two GCs, 43% for the Eight Puzzle, and 89% for navigation (reducing it to 391 MC/GC). Total speedup factors are thus seen that range from 2.5 to 10 across these three tasks. Concurrently, msec/MC has stayed roughly the same for semantic memory, at .1, while for the other two tasks it has dropped from .6 to .5, providing an additional speedup factor of 1.2 for these harder problems. With a new maximum of 602 MC/GC over these three tasks (for the Eight Puzzle), msec/GC would now need to be .08 – a factor of 6.25, rather than the original 46, from the current maximum of .5 – to enable all three tasks to proceed within 50 msec/GC.

(Simulated) Parallelism

Instead of reducing the number of message cycles by reducing the number of messages that need to be sent, parallelism enables multiple messages to be sent within each message cycle. One simple form of this is to send messages out in parallel along each active direction of each link of the graph, as long as there is a new message there to be sent. With such an approach, msec/GC becomes the product of msec/MC and the number of parallel message cycles (MC). In the absence of loops, MC should be bounded by the depth of the graph, again implying that the structure of the graph – in particular, how messages on deeper links depend on those on shallower links – is critical. With loops, there is no obvious a priori bound.

Although the architecture has not yet been ported to parallel hardware, a message-passing discipline has been implemented that is based on a sequence of (simulated) parallel message cycles. The first optimization introduced above, of reusing messages across graph cycles, may still be relevant with parallel message cycles; however, the second is not, given that all queued messages are effectively sent during each parallel message cycle. Although this form of parallelization implies that more total messages may be sent, sending them in parallel may radically reduce MC/GC while keeping msec/MC nearly the same.

With parallel message passing turned on and no message reuse across graph cycles, the average number of messages per cycle rises to 658 for semantic memory, 2758 for the Eight Puzzle, and 3747 for navigation; yet, the average MC/GC is only 26, 33, and 76 for the three tasks. MC/GC turns out to be relatively stable within each of these tasks, with navigation running a constant 76 and the Eight Puzzle ranging from a low of 29 to a high of 36. Given a maximum of 76 MC/GC across these three tasks, 50 msec/GC becomes feasible with an msec/MC of .7. If the communication overhead on parallel hardware is a small fraction of this, the existing maximum of .5 msec/MC should be sufficient to yield a real-time graph cycle. Such an approach also has the advantage of removing the need for a global queue, enabling message passing to be truly local.

When (simulated) parallelism is combined with message reuse across graph cycles, the average number of messages

per GC remains at 658 for semantic memory, but drops to 1962 for the Eight Puzzle and 3637 for navigation, yielding reductions of 29% and 3% for these latter two. The average MC/GC becomes 26, 28, and 76 for the three tasks, a 15% gain for the Eight Puzzle but no change for the other two.

Conclusion

With serial message passing, the first two optimizations introduced here reduce MC/GC across semantic memory, the Eight Puzzle and a navigation task by a factor of 2.5-10. Given that the optimizations also reduced the time per message cycle for the harder problems by a factor of 1.2, the total gain in time per cognitive cycle is a factor of 3-12. When considering the worst case over these three tasks, an additional factor of 6.25 is now needed to achieve 50 msec per cognitive cycle, a significant improvement over the factor of 46 that was needed at the start.

Parallelization provides a somewhat different approach to reducing MC/GC, by sending messages in parallel within message cycles. If close to the full amount of potential parallelism can be achieved on parallel hardware, it provides a path, albeit a more costly one in terms of hardware, for immediately reaching the 50 msec threshold. Even on a workstation with 2-8 cores, it may be able to help significantly in reaching this threshold, particularly if some form of the message ordering optimization were able to eliminate messages that don't really need to be sent within early message cycles (which tend to be the most computationally intensive).

For the future, it will be important to explore whether message reuse across graph cycles can be extended to a larger fraction of the graph, whether there is an analogue of the node-depth optimization that works for loopy graphs, and what would happen with a deployment on true parallel hardware. It is also important to investigate what additional gains may be had in terms of msec/MC, where a sparse function representation is currently being explored, but where other possibilities also exist. It may also ultimately prove worthwhile to consider switching from summary product to algorithms that are more approximate, based on sampling, particle filters, or variational methods. This may become particularly critical as the task complexity continues to scale up in various ways.

Acknowledgements

This work has been sponsored by the U.S. Army. Statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

References

- Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* Oxford: Oxford University Press.
- Card, S. K., Moran, T.P. & Newell, A. (1983), *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Chen, J., Demski, A., Han, T., Morency, L-P., Pynadath, P., Rafidi, N. & Rosenbloom, P. S. (2011). Fusing symbolic and decision-theoretic problem solving + perception in a graphical cognitive architecture. *Proceedings of the 2nd International Conference on Biologically Inspired Cognitive Architectures*.
- Derbinsky, N., Laird, J. E. & Smith, B. (2010). Towards efficiently supporting large symbolic declarative memories. *Proceedings of the 10th International Conference on Cognitive Modeling*.
- Doorenbos, R. B. (1993). Matching 100,000 rules. *Proceedings of the 11th National Conference on Artificial Intelligence*.
- Douglass, S. A. & Myers, C. W. (2010). Concurrent knowledge activation calculation in large declarative memories. *Proceedings of the 10th International Conference on Cognitive Modeling*.
- Forgy, C. L. (1982). Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem. *Artificial Intelligence*, 19, 17-37.
- Kieras, D. E. & Meyer, D. E. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction*, 12, 391-438.
- Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA: MIT Press.
- Kschischang, F. R., Frey, B. J. & Loeliger, H. (2001). Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47, 498-519.
- Laird, J. E. (2012). *The Soar Cognitive Architecture*. Cambridge, MA: MIT Press. In press.
- Rosenbloom, P. S. (2010). Combining procedural and declarative knowledge in a graphical architecture. In *Proceedings of the 10th International Conference on Cognitive Modeling*.
- Rosenbloom, P. S. (2011a). Rethinking cognitive architecture via graphical models. *Cognitive Systems Research*, 12, 198-209.
- Rosenbloom, P. S. (2011b). Bridging dichotomies in cognitive architectures for virtual humans. *Proceedings of the AAAI Fall Symposium on Advances in Cognitive Systems*.
- Rosenbloom, P. S. (2011c). From memory to problem solving: Mechanism reuse in a graphical cognitive architecture. *Proceedings of the Fourth Conference on Artificial General Intelligence*.
- Rosenbloom, P. S. (2011d). Mental imagery in a graphical cognitive architecture. *Proceedings of the Second International Conference on Biologically Inspired Cognitive Architectures*.
- Tambe, M. & Rosenbloom, P. S. (1994). Investigating production system representations for non-combinatorial match. *Artificial Intelligence*, 68, 155-199.

Practical Optimal-Solution Algorithms for Schema-based Analogy Mapping

Robert Hamilton (rhamilton@mun.ca) and Todd Wareham (harold@mun.ca)

Department of Computer Science
Memorial University of Newfoundland
St. John's, NL Canada A1B 3X5

Keywords: Analogy Mapping; Computational Complexity

Schema-based Analogy Mapping

In the last 20 years, analogy derivation has come to be at the forefront of cognitive science (Gentner, 2010). Under Structure Mapping Theory (Gentner, 1983), an **analogy** “ T is (like) a B ”, where B and T are predicate-structures, is a mapping from a portion of B to a portion of T that satisfies various conditions (see Figure 1). This yields the following computational problem:

ANALOGY MAPPING

Input: Two predicate-structures B and T .

Output: The most systematic analogy mapping M between B and T .

One popular heuristic for making this problem simpler is to use schemas. While there is no formal definition, a schema is described in Gentner et al. (2009) as “... the relational structure engendered by an analogical comparison ... [which] will be a fairly concentrated relational representation, with many of the initial item-specific features stripped away” (p. 1345). It has been claimed that schema-based analogy derivation is efficient in practice, i.e.

... aligning a target with [a schema] should be computationally less costly than aligning a target with the corresponding literal base concept [because schemas] will contain fewer predicates than the literal concepts they were derived from, and a higher proportion of these predicates can be mapped to relevant target concepts. (Bowdle & Gentner, 2005, p. 199)

The common thread in such claims is that schemas make analogy derivation easier because schemas are small and will generally be fully or almost-fully mapped to a comparison predicate-structure. However, these claims have never been formally proven. In this poster, we give preliminary results of a complexity-theoretic investigation of these claims.

Methodology

First, we establish the complexity of ANALOGY MAPPING. Then, we analyze a simplified version of this problem dealing specifically with schemas in order to find efficient algorithms. Following convention in Computer (Garey & Johnson, 1979) and Cognitive (van Rooij, 2008) Science, an algorithm is considered efficient if it runs in polynomial time, i.e. in time upper-bounded by n^c where n is the input size and c is a constant. It is widely held that no such algorithm exists

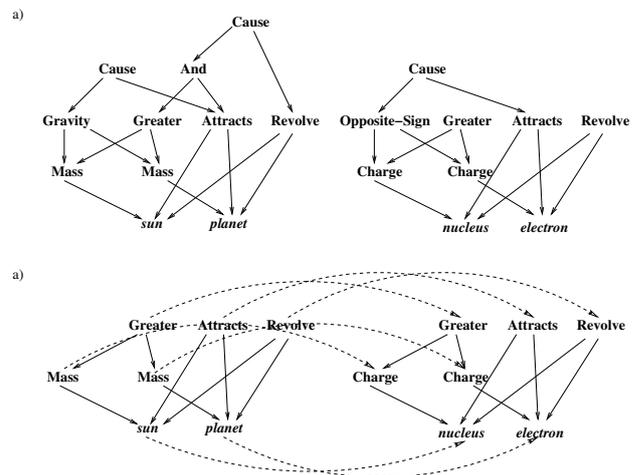


Figure 1: Analogy Derivation in Structure-Mapping Theory. (a) Two graph representations of predicate structures encoding descriptions of the solar system (left) and the Rutherford model of the atom (right). (b) An analogy between the structures in (a), where dotted arrows indicate the mappings between corresponding pairs of predicates and objects.

for a problem if that problem is NP -hard. In such cases, we consider two ways of restricting the problem input to allow for a practical solution:

- Restricting the values of a set K of one or more problem-aspects (**parameters**) such that there are algorithms that are **fp-tractable** for those parameters, i.e., algorithms that run in time $f(K)n^c$ for some function f , and hence are effectively polynomial-time when those parameters are restricted (Downey & Fellows, 1999).
- Limiting the inputs to certain classes of graphs. The classes considered here are directed trees (DT), polytrees (PT) (directed acyclic graphs which remain acyclic even if the direction of their arcs is removed), polyforests (PF), and directed acyclic graphs (DAG).

Complexity Results

For reasons of space, all proofs are omitted; they can be found in Hamilton (2012). It is known that ANALOGY MAPPING is NP -hard (van Rooij et al., 2008) and hence, modulo the widely-believed conjecture that $P \neq NP$ (Fortnow, 2009), cannot be solved efficiently in general. Let $|T|$ be the size of the target graph T and d be the difference in size between T and the optimal analogy mapping.

Lemma 1 ANALOGY MAPPING is *fp-intractable* for $\{d, |T|\}$.

This shows that the properties of size and closeness that schemas possess are not sufficient on their own to guarantee efficiency. For this reason, we examine inclusion, a special case where d must be zero.

(i, j) C-ANALOGY INCLUSION $[(i, j)$ C-AI]

Input: Two ordered predicate-structures B and T of class C with i and j roots, respectively.

Output: An analogy mapping M between B and T such that T is completely mapped onto B , or \perp if no such mapping exists.

Note that we also restrict predicate-structures to consist of predicates whose arguments are ordered, e.g., GREATER(X, Y). The number of roots is of special interest, as the only optimal-solution algorithm known for ANALOGY MAPPING (Falkenhainer et al., 1989) exhibits improved performance when the number of roots is restricted.

Relative to the various classes of predicate-structure graphs mentioned previously, we have the following results:

Lemma 2 $(1, 1)$ DT-AI can be solved in $O(|T|)$ time.

Lemma 3 (i, j) PT-AI can be solved in $O(|T|^{1.5}|B|^2)$ time.

Lemma 4 (i, j) PF-AI is NP-Hard.

The frontier of general practicality for ANALOGY INCLUSION is thus polyforests. At this point, we must examine possible parameters to make this case solvable efficiently. Recall that j is the number of roots in T and let f be the maximum number of occurrence of any root predicate-type in B or T .

Lemma 5 (i, j) PF-AI is *fp-tractable* for $\{f, j\}$.

As polyforests are special cases of DAGs, this result also holds for (i, j) DAG-AI. Moreover, as ANALOGY INCLUSION is a special case of ANALOGY MAPPING, all *fp-tractability* results and most of the *fp-intractability* results for ANALOGY MAPPING given in van Rooij et al. (2008) and Wareham et al. (2011) hold for (i, j) DAG-AI as well.

Discussion

In this poster, we have shown that the frontier of polynomial-time tractability for schema-based analogy mapping is in fact lower than general DAGs and given a rough assessment of the *fp-tractability* options for such mapping relative to polyforests and DAGs. Much work remains to be done, both to establish the complexity of ANALOGY INCLUSION relative to all combinations of the considered parameters and to extend these results back to general schema-based analogy mapping.

There are also closely-related problems of interest. For example, it has been conjectured that ANALOGY MAPPING is easier when both B and T (rather than only T to B) are close (Gentner, 2010). The limiting case analogous to ANALOGY INCLUSION is determining the mapping between analogically identical predicate-structures. While we do not know

the complexity of this problem, we do have results for a related problem, namely IDENTICAL ANALOGY, which returns “yes” if predicate-structures B and T are isomorphic, i.e. is there an analogy mapping between all of B and all of T ?

Lemma 6 IDENTICAL ANALOGY is *polynomial-time equivalent* to GRAPH ISOMORPHISM.

As GRAPH ISOMORPHISM is widely believed to be polynomial-time intractable, this result provides circumstantial evidence that mutual predicate-structure closeness is not a sufficient restriction to make ANALOGY MAPPING efficiently solvable, and hence motivates the application of the methodology described here to this problem.

References

- Bowdle, B., & Gentner, D. (2005). The career of metaphor. *Psychological Review*, 112(1), 193–216.
- Downey, R., & Fellows, M. (1999). *Parameterized Complexity*. Berlin: Springer.
- Falkenhainer, B., Forbus, K. D., & Gentner, D. (1989). The Structure-Mapping Engine: Algorithm and examples. *Artificial Intelligence*, 41, 1–63.
- Fortnow, L. (2009). The Status of the P Versus NP Problem. *Communications of the ACM*, 52(9), 78–86.
- Garey, M. R., & Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: W.H. Freeman.
- Gentner, D. (1983). Structure-Mapping: A Theoretical Framework for Analogy. *Cognitive Science*, 7, 155–170.
- Gentner, D. (2010). Bootstrapping the mind: Analogical processes and symbol systems. *Cognitive Science*, 34(5), 752–775.
- Gentner, D., Lowenstein, J., Thompson, L., & Forbus, K. D. (2009). Reviving Inert Knowledge: Analogical Abstraction Supports Relational Retrieval of Past Events. *Cognitive Science*, 33(6), 1343–1382.
- Hamilton, R. (2012). *Practical Optimal-Solution Algorithms for Schema-based Analogy Mapping*. Honours thesis, Department of Computer Science, Memorial University of Newfoundland.
- van Rooij, I. (2008). The Tractable Cognition Thesis. *Cognitive Science*(32), 939–984.
- van Rooij, I., Evans, P., Müller, M., Gedge, J., & Wareham, T. (2008). Identifying Sources of Intractability in Cognitive Models: An Illustration using Analogical Structure Mapping. In *Proceedings of the 30th Annual Conference of the Cognitive Science Society* (pp. 915–920). Austin, TX: Cognitive Science Society.
- Wareham, T., Evans, P., & van Rooij, I. (2011). What Does (and Doesn't) Make Analogical Problem Solving Easy? a Complexity-Theoretic Investigation. *Journal of Problem Solving*, 3(2), 30–71.

Extending the Computational Belief-Desire Theory of Emotions to Fantasy Emotions

Rainer Reisenzein (rainer.reisenzein@uni-greifswald.de)

Institute of Psychology, University of Greifswald
Franz-Mehring-Straße 47, 17489 Greifswald, Germany

Keywords: Computational belief-desire theory of emotion, affective computing, fantasy emotions, imaginative emotions, emotional reactions to fiction, simulation theory, paradox of fiction.

Literary and cinematic depictions of events frequently evoke emotions in readers or viewers even when they are fully aware that the portrayed events are fictitious. Likewise, the deliberate imagination of counterfactual events can evoke emotions. These “fantasy emotions”, as the Austrian philosopher-psychologist Meinong (1910) called them, pose an explanatory problem for cognitive emotion theories. The reason is that these theories assume—at least in their standard form—that emotions presuppose *beliefs* in the existence of the emotion-eliciting events; but such beliefs seem to be lacking in the case of the fantasy emotions (e.g., Green, 1992). In philosophy of art, these considerations have given rise to the much-discussed “paradox of emotional response to fiction” (Radford, 1975; Schneider, 2011). To solve the problem, Meinong (1910) proposed that the fantasy emotions are not based on beliefs but on a different kind of cognitive propositional attitude, called *assumptions* (*Annahmen*). My aim is to explicate Meinong’s theory of fantasy emotions in the context of CBDTE, a (sketch of a) computational (C) model of the belief-desire theory of emotion (BDTE) (see Reisenzein, 2009a; 2009b).

The Belief-Desire Theory of Emotion

BDTE is a version of cognitive emotion theory (see e. g., Marsella, Gratch, & Petta, 2010). Its basic assumption is that the core set of the mental states presystematically called “emotions” presuppose, for their existence, both *beliefs* (cognitive or informational states) and *desires* (motivational states) about the emotion-eliciting states of affairs. Thus, the conceptual framework of BDTE is the same as that of the belief-desire theory of action that inspired the BDI (belief-desire-intention) approach to artificial agents (e.g., Bratman, Israel, & Pollack, 1988; Hindriks, 2009). More precisely, emotions are reactions to the cognized actual or potential fulfillment or frustration of desires; plus, in some cases (e.g., relief, disappointment), the confirmation or disconfirmation of beliefs (Reisenzein, 2009a; 2009b). To illustrate, Mary is *happy* that *p* (e.g., that *Mr. Schroiber was elected chancellor*) if she desires *p* and now comes to believe firmly (i.e., is certain) that *p* is the case; whereas Mary is *unhappy* that *p* if she is averse to *p*, and now comes to believe firmly that *p* is the case.

CBDTE: A Computational Explication of BDTE

Following Fodor (1987), CBDTE (Reisenzein, 2009a; 2009b) assumes that beliefs and desires (the causes of emotion according to BDTE) are special modes of processing propositional representations, i.e. sentences in a “language of thought”. It is assumed that the central part of this propositional representation system is innate and that its innate components comprise a set of hardwired maintenance and updating mechanisms. At the core of these mechanisms are two comparator devices, the belief-belief comparator (BBC) and the belief-desire comparator (BDC). The BBC compares newly acquired beliefs to pre-existing beliefs, whereas the BDC compares them to existing desires. Computationally speaking, the BBC and BDC compare the “mentalese” sentence tokens s_{new} representing the contents of newly acquired beliefs, with the sentences s_{old} representing the contents of pre-existing beliefs and desires. If either a match (s_{new} is identical to s_{old}) or a mismatch (s_{new} is identical to $not-s_{old}$) is detected, the comparators generate an output that communicates the detection and degree of the match or mismatch to the rest of the cognitive system. CBDTE assumes that the comparator mechanisms operate automatically (without intention, and preconsciously) and that their outputs are *nonpropositional*: They consist of signals that vary in kind and intensity, but have no internal structure, and hence are analogous to sensations (e.g., of tone or temperature). Output signals that exceed a certain threshold of intensity give rise, directly or indirectly, to unique conscious feeling qualities: the feelings of surprise and expectancy confirmation (BBC), and the feelings of pleasure and displeasure (BDC). According to CBDTE, the BDC and BBC are the basic emotion mechanisms of humans.

Fantasy Emotions in CBDTE

Meinong (1910) proposes that *assuming* is a special mode of cognitively representing states of affairs: the person posits, or hypothetically supposes, that *p* is the case. Furthermore, he suggests that whereas serious emotions are based on beliefs, fantasy emotions are based on assumptions. In the framework of BDTE, this suggestion can be interpreted as follows: One experiences serious joy about *p* if one desires *p* and believes (or more precisely, comes to believe) that *p* is the case; whereas one experiences fantasy joy about *p* if one desires *p* and *assumes* that *p* is the case (Reisenzein, 2012).

To incorporate fantasy feelings into CBDTE, I begin by assuming that, like believing *p* and desiring *p*, *assuming p* is a special mode of processing propositional representations.

An elaboration of this idea has been proposed by Nichols and Stich (2003) in their theory of mental simulation. However, to explain fantasy emotions, important extensions of this model are needed. These extensions are directly suggested by CBDTE's assumptions about serious emotions. Specifically, I assume that the updating mechanisms for assumptions include hardwired comparator mechanism analogous to the BBC and BDC: An *assumption-assumption comparator* (AAC), and an *assumption-desire comparator* (ADC). The AAC compares newly made assumptions with existing assumptions, whereas the ADC compares newly made assumptions with existing desires. Fantasy emotions arise when the AAC or the ADC detect an agreement or a conflict between (a) a newly made assumption and (b) an existing assumption or desire, respectively.

To illustrate, Mary experiences fantasy joy about Schroiber's election victory ($= p$) if she desires p and assumes p to be the case. On the computational level, this corresponds to: Mary's ADC discovers that the mental sentence representing the content of an existing desire is identical to that of a newly made assumption; as a consequence, it generates a nonpropositional signal that communicates the detection of this agreement to the rest of the cognitive system, and that is subjectively experienced as a feeling of fantasy pleasure. Analogously, Mary experiences *fantasy displeasure* about p if she is averse against p and assumes p to be the case. On the computational level, this corresponds to: Mary's ADC discovers that there is a contradiction between the content of an existing desire and the newly made assumption p ; it then generates a signal which communicates the detection of this incongruence to the rest of the cognitive system, and which is subjectively experienced as fantasy displeasure. Mary can also experience *fantasy surprise*—namely, if she first assumed that Schroiber did not win the election ($\text{not-}p$) and then makes the new assumption that Schroiber did, after all, win the election (p). In this case, Mary's AAC detects a contradiction between an assumption that is part of a current simulation and a newly made assumption, and as a consequence generates a signal that is experienced as fantasy surprise.

Explanatory Capacity of the Theory

CBDTE can explain the thorough-going *parallelism* between fantasy feelings and serious feelings. Each serious emotion (joy, sorrow, fear, hope etc.) can also occur in a fantasy form (as fantasy joy, fantasy sorrow, and so on). Likewise, both serious and fantasy emotions can be experienced in different intensities and both can be directed at the same state of affairs. According to CBDTE, this parallelism between serious and fantasy emotions is the consequence of the parallel construction of their generating mechanisms. CBDTE can also account for the different motivational effects of serious and fantasy emotions (see Schneider, 2011): Whereas serious emotions often motivate coping actions, the corresponding fantasy emotions usually do not have such effects. CBDTE can explain this difference, at least in part, by the assumption that an immediate update of beliefs and desires takes place only in the case of serious emotions, but not in the case of the fantasy emotions. Fantasy emo-

tions can influence actions only indirectly; in particular, by generating beliefs *about* fantasy emotions. Finally, the CBDTE theory of fantasy emotions throws new light on the question of whether or not fantasy emotions qualify as "genuine" emotions (Reisenzein, 2012).

References

- Bratman, M., Israel, D., & Pollack, M. (1988). Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4, 349–355.
- Fodor, J. A. (1987). *Psychosemantics: The problem of meaning in the philosophy of mind*. Cambridge, MA: MIT Press.
- Green, O. H. (1992). *The emotions: A philosophical theory*. Dordrecht: Kluwer.
- Hindriks, K. V. (2009). Programming rational agents in GOAL. In R. H. Bordini, M. Dastani, J. Dix, & A. El Fallah Seghrouchni (Eds.), *Multiagent Programming: Languages, tools and applications* (pp. 119–157). Berlin: Springer.
- Marsella, S., Gratch, J., & Petta, P. (2010). Computational models of emotion. In K. R. Scherer, T. Bänziger, & E. Roesch, (Eds.) *A blueprint for a affective computing: A sourcebook and manual*. Oxford: Oxford University Press.
- Meinong, A. (1910). *Über Annahmen [On Assumptions]*. (2nd Ed.) Leipzig, Germany: Barth. Reprinted in: R. Haller & R. Kindinger (Eds.) (1977). *Alexius Meinong Gesamtausgabe [Alexius Meinong's complete works]*. (Vol. 4). Graz, Austria: Akademische Druck- und Verlagsanstalt.
- Nichols, S., & Stich, S. (2003). *Mindreading: An integrated account of pretense, self-awareness and understanding other minds*. Oxford: Oxford University Press.
- Radford, C. (1975). How can we be moved by the fate of Anna Karenina? *Proceedings of the Aristotelian Society*, Supplemental Vol. 49, 67–80.
- Reisenzein, R. (2009a). Emotions as metarepresentational states of mind: Naturalizing the belief-desire theory of emotion. *Cognitive Systems Research*, 10, 6–20.
- Reisenzein, R. (2009b). Emotional experience in the computational belief-desire theory of emotion. *Emotion Review*, 1, 214–222.
- Reisenzein, R. (2012). Fantasiegefühle aus der Sicht der kognitiv-motivationalen Theorie der Emotion [Fantasy feelings from the perspective of the cognitive-motivational theory of emotion]. In S. Poppe (Ed.), *Emotionen in Literatur und Film* (pp. 31–63). Würzburg: Königshausen & Neumann.
- Schneider, S. (2011). The paradox of fiction. *Internet Encyclopedia of philosophy*. <http://www.iep.utm.edu/fict-par/> (retrieved June 17, 2011).

Dual-route Connectionist Model of Greek Spelling

Ioanna Katidioti (ikatiديو@phs.uoa.gr)

Graduate Program in Cognitive Science, Athens University Campus
GR-15771 Athens, Greece

Ian C. Simpson (isimpson@ugr.es)

Departamento de Psicología Evolutiva y de la Educación, Campus de Cartuja s/n
18071 Granada, Spain

Athanassios Protopapas (aprotopapas@phs.uoa.gr)

Department of Philosophy & History of Science, University of Athens
GR-15771 Athens, Greece

Keywords: connectionist model; spelling; phoneme-grapheme mapping

Introduction

We created a dual-route connectionist model of Greek spelling. The model maps sequences of phonemes to corresponding sequences of graphemes, using a sublexical and a lexical route, i.e., phonographemic information and word knowledge, respectively. It is based on the model of Houghton and Zorzi (2003), but handles words up to 5 syllables long, with full connectivity between the syllables. Greek has 37 phonemes and 84 graphemes related via 118 mappings with 80,3% consistency (spelling) (Protopapas & Vlahou, 2009). Model architecture is as follows:

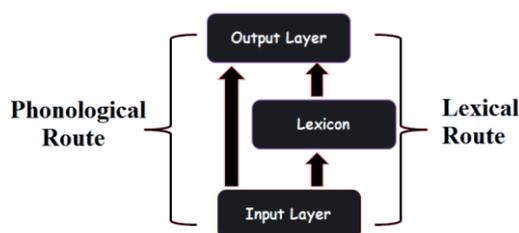


Figure 1: Dual-route model of spelling

Input-Output Representation

The representation is syllabic and nucleus-centered. There are 4 consonant slots on each side of the vowel. The orthographic slots are occupied by graphemes, not letters.

Input-Output Representation

Input (Phonological form)

§ _ _ _ i _ _ _ _	b _ _ _ ' e _ _ _	k t _ _ e s _ _	_____	_____
cccc v cccc				
1 st syllable	2 nd syllable	3 rd syllable	4 th syllable	5 th syllable

Output (Orthographical form)

σ _ _ v _ _ _	μ π _ _ α ι _ _	κ τ _ _ ε ς _ _	_____	_____
cccc v cccc				
1 st syllable	2 nd syllable	3 rd syllable	4 th syllable	5 th syllable

Figure 2: Input and output representation

Training and parameters

To simulate spelling development using children's data, we trained the model to a corpus of 30,391 words from elementary school books. The model was trained for 30 epochs, with learning rate 0.02 and no weight pruning. During spelling, feedback was set to a value of 0.2.

Results

Using both routes, the entire training set is spelled correctly. Using only the phonological route, 65.2% of the training set is spelled correctly and almost all errors are phonologically plausible. By adding a small contribution from the lexical route we were able to simulate Grade 3-4 children's data of 48 words. In the simulation, 13 out of 14 mistakes were the same as those made by the children, and 11 of these were the most typical.

Problems

The model made two kinds of phonologically implausible mistakes: it spelled /s/ inside 19 words with "ς" (which is only used word-finally) and it also omitted a grapheme in a few words. In addition, the model has two problems: (a) the number of cycles needed to compute the output don't always correspond to the difficulty of the word and (b) certain palatal consonants were consistently misspelled (e.g. /ca/ as "κα" instead of "κια").

Empirical validation

Greek has a number of ambiguous phonemes, the alternative spellings of which appear with different frequency (Protopapas & Vlahou, 2009). For example, in our training corpus, the phoneme /o/ is spelled with the letter "ο" 74% of the time and with "ω" 26%. Due to frequency-sensitive training the model usually spells the ambiguous phonemes with the highest-frequency grapheme. However, due to asymmetries in the distribution of consonant-vowel co-occurrences, this is not always the case. That is, the model will use the less frequent graphemic variant of a phoneme when more likely in the particular phonographemic context. If the model corresponds to human spelling performance,

children should also be more likely to choose the less frequent graphemes in the same contexts.

To test this prediction, we created two groups of 39 nonwords each, with ambiguous phonemes (o, e, i and g). Group A included nonwords spelled by the model with a low-frequency grapheme (“ω”, “α”, “οι”, “υ” and “γγ”). This was accomplished by inspection of the model’s weights, choosing consonants with strong weight connections to target graphemes. Group B included similar nonwords (same number of phonemes and consonant-vowel structure) that were spelled by the model with the high-frequency graphemic alternative (“ο”, “ε”, “ι” and “γκ”). For example, nonwords /xo’θafo/ and /mo’ðamo/ were spelled by the model as “χωθάφο” and “μοδάμο”, respectively (note the ω/o difference in the second position).

Participants

177 students of the elementary Grades 5-6 participated in the experiment. Each child spelled 39 nonwords dictated by the experimenter.

Results

The relative proportion of frequent vs. infrequent grapheme used by the children in each nonword group was examined for each phoneme using generalized linear mixed-effects models in R (function lmer of package lme4). The interaction of item group (A vs. B) by grapheme frequency (high vs. low) was significant in every case (i.e., for every phoneme tested), indicating that participants wrote more Group A items with a low-frequency grapheme than Group B items.

- For /o/ (ο-ω): $\beta = -2.87$ $z = -4.7$, $p < .0005$
- For /e/ (ε-α): $\beta = 2.22$ $z = 4.77$, $p < .0005$
- For /i/ (ι-οι): $\beta = -2.04$ $z = -2.82$, $p = .005$
- For /i/ (ι-υ): $\beta = -2.12$ $z = -4.62$, $p < .0005$
- For /g/ (γκ-γγ): $\beta = .76$ $z = 2.92$, $p = .004$

Discussion

The model spells known words perfectly, based on the lexical route. When only the phonological route is used, almost all errors are phonologically plausible. The model also simulates children’s data successfully. We created nonwords using the model’s weights in order to promote the use of low-frequency graphemes for ambiguous phonemes. Children were influenced by the context of ambiguous phonemes, which indicates that the frequency of phoneme-grapheme co-occurrence affects spelling. In conclusion, our model is a useful tool for exploring the development and difficulties of Greek spelling

References

Houghton, G. & Zorzi, M. (2003). Normal and impaired spelling in a connectionist dual-route architecture. *Cognitive Neuropsychology*, 20, 115–162.

Mouzaki, A., & Protopapas, A. (Eds.) (2010) *Ορθογραφία: Μάθηση και διαταραχές [Spelling: Learning and disorders]*. Athens: Gutenberg.

Protopapas, A., & Vlahou, E. L. (2009). A comparative quantitative analysis of Greek orthographic transparency. *Behavior Research Methods*. 41, 991–1008.

Matching Results of Latent Dirichlet Allocation for Text

Andreas Niekler (aniekler@fbm.htwk-leipzig.de)

Leipzig University of Applied Sciences (HTWK), Faculty of Media
Karl-Liebknecht-Str. 145, 04277 Leipzig, Germany

Patrick Jähnichen (jaehnichen@informatik.uni-leipzig.de)

Natural Language Processing Group, Department of Computer Science
University of Leipzig
Johannisgasse 26, 04103 Leipzig, Germany

Abstract

Many approaches have been introduced to enable Latent Dirichlet Allocation (LDA) models to be updated in an on-line manner. This includes inferring new documents into the model, passing parameter priors to the inference algorithm or a mixture of both, leading to more complicated and computationally expensive models. We present a method to match and compare the resulting LDA topics of different models with light weight easy to use similarity measures. We address the on-line problem by keeping the model inference simple and matching topics solely by their high probability word lists.

Keywords: Latent Dirichlet Allocation, topic distance measures, on-line topic tracking

Introduction

As massive amounts of information become available on-line, text mining applications have become an integral part of both industry and academia. One field of text mining is the identification and extraction of semantic concepts in text documents. Over the last decade, Latent Dirichlet Allocation (D. M. Blei, Ng, & Jordan, 2003) (LDA) has become one of the most popular methods to approach this task. LDA is a Bayesian model that makes use of latent variables¹, which represent the semantic concepts (associated with LDA and models building on LDA, these concepts are known as *topics*), to compute the posterior probability over the latent variables and model parameters to allow the extraction of latent semantic structures in texts (i.e. the topics). Examination of the posterior allows an approximation of probability distributions for both documents and topics².

Having a technique at hand to identify different topics, the wish to study their evolution over time evolves naturally. This includes both the analysis of static corpora as well as data that comes in constantly via a stream, the latter of which mostly relies on the segmentation of the data into different time slices of predefined size (e.g. one hour, one day, one year etc.), treating newly arrived data as a new time slice after its size is reached. Another way of handling the data and tracking topic trends without segmentation into time slices is that introduced by (Wang & McCallum, 2006), where the authors use the time stamps of documents as an additional (continuous) observed variable in the model. However, in our approach we resort to the notion of time slice separated data.

¹For an introduction to latent variable models see (Bishop, 1999)

²For documents, a probability distribution over the set of latent topics and analogous to that, for topics, a probability distribution over a fixed vocabulary is inferred

The main problem of tracking topics' evolutions over time, either statically or in an on-line manner is the identification of identical topics in consecutive time slices or data windows³. To overcome this, previous approaches such as (D. Blei & Lafferty, 2006; AlSumait, Barbará, & Domeniconi, 2008) use the model outcome of time $t - 1$ as a prior for the model at time t or analogously the outcome of a data window as a prior for another sub-set of documents. As this is rather an elegant way to align topics between time slices (from a mathematical point of view), these methods suffer from two serious drawbacks concerning the analysis of diachronic document collections. First, those models are restricted to use the same number and effectively the same topics in each time slice and are bound to measure the amount of change a specific topic undergoes from time $t - 1$ to t instead of just aligning possibly identical topics. This prevents from finding newly arising and also from releasing "died", i.e. now unused topics or could even lead to the connection of unrelated topics. Second, the approach of using the outcome of the model at time $t - 1$ (or a data window) as an input for the model at time t (or another set of documents) forces the analysis to be processed in a one-after-another fashion, preventing parallel processing of the data.

For the sake of completeness, it shall be stated that another approach to this field is known as *Topic Detection and Tracking*, for which (Allan, 2002) gives a detailed introduction.

In this paper we propose the matching of topics from subsequently trained LDA models via lightweight statistical similarity measures. Our approach is motivated by the finding that a major probability mass in topics' distributions over a vocabulary is represented only by a small number of highly probable words in the distribution. We therefore restrict ourselves to using only a subset of words, together with their probabilities to match different topics. This enables us to independently train the LDA models on each time slice, including both parameter and number of topics optimization per time slice. Further enhancements, such as using hierarchical Bayesian models (e.g. the hierarchical Dirichlet process model introduced by (Teh, Jordan, Beal, & Blei, 2006)) instead of optimizing the number of topics per time slice, are possible without altering the approach.

³A sub-set of documents from a bigger corpus.

The paper is organized as follows. In section 2, we review the underlying LDA model and describe our approach for matching topics of different time slices in section 3. Section 4 relates our approach to previous ones and subsumes them. We give an overview over the different similarity measures we took into consideration for solving the task in section 5 and present experiments and results in sections 6 and 7 using hand selected topics generated from a document corpus from the UK-based newspaper *The Guardian*, collected through an API on consecutive days from March, 10th through March, 15th 2011. Finally, we conclude giving an outlook to possible applications and future work.

LDA Model

Before defining our approach for matching topics, we first give a review of the statistical model of LDA and a Gibbs sampling algorithm introduced by (Griffiths & Steyvers, 2004), as a method for inference in the model. LDA is a hierarchical Bayesian model that encodes the relation between words and documents via the latent topics in a document corpus. Herein, documents are not directly linked to words but through latent variables z that govern the responsibility of a certain topic for the words in a document. As words are the observable variables in this model, conditional independence holds true for the document and topic distributions θ and ϕ . Placing prior distributions with hyperparameters α and β over θ and ϕ respectively completes the probabilistic model. A generative process for document generation is given by

1. draw K multinomials $\phi_k \propto \text{Dir}(\beta_k)$, one for each topic k
2. for each document $d, d = 1, \dots, D$
 - (a) draw multinomial $\theta_d \propto \text{Dir}(\alpha_d)$
 - (b) for each word w_{dn} in document $d, n = 1, \dots, N_d$
 - i. draw a topic $z_{dn} \propto \text{Multinomial}(\theta_d)$
 - ii. draw a word w_{dn} from $p(w_{dn}|\phi_{z_{dn}})$, the multinomial probability conditioned on topic z_{dn}

Exact inference is not tractable in this model, thus we utilize Gibbs sampling as described by (Griffiths & Steyvers, 2004). This includes computing the posterior distribution over all variables and model parameters instead of inferring θ and ϕ directly. Examination of the posterior then yields both distributions. The posterior distribution over topic assignments to words, conditioned on the words and all other topic assignments is given by

$$p(z_i = j | \mathbf{z}_{\setminus i}, \mathbf{w}) \propto \frac{C_{w_{\setminus i}, j}^{VK} + \beta_{w_i}}{\sum_{v=1}^V (C_{v_{\setminus i}, j}^{VK} + \beta_w)} \frac{C_{d_{\setminus i}, j}^{DK} + \alpha_j}{\sum_{k=1}^K (C_{d_{\setminus i}, k}^{DK} + \alpha_k)} \quad (1)$$

where C^{VK} and C^{DK} are count matrices with dimensions $V \times K$ and $D \times K$, representing the number of times, a word has been assigned to a topic and the number of times, a topic has been assigned to a document, respectively. Subscript $\setminus i$ excludes the current assignment. Both matrices can be stored

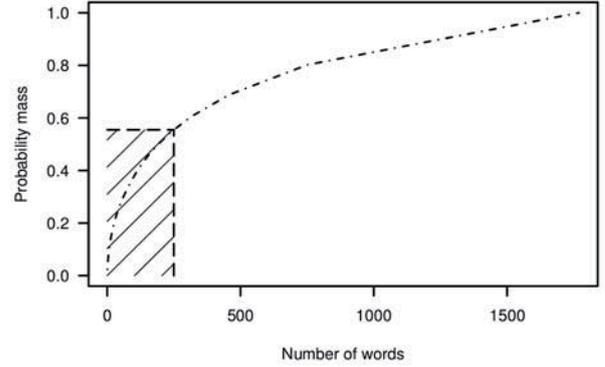


Figure 1: CDF plot for sorted $p(\mathbf{w}|z_k)$ probability distribution example

efficiently, using a sparse matrix representation, allowing a large vocabulary and thus large document corpora to be processed. Examination of the posterior leads to approximations of both ϕ and θ , which are given as the first and second fraction of equation (1). Consequently, ϕ can be interpreted as a matrix of size $V \times K$, containing the conditional probability $p(w_i|z_k)$ at position $\phi_{i,k}$. Hence, every column vector of ϕ , $\phi_{\cdot,k}$ can be construed as a probability distribution over the whole vocabulary of size V for topic k . The row vectors $\theta_{k,\cdot}$ of matrix θ with $\theta_{k,m} = p(z_k|d_m)$ can then be seen as probability distributions over all latent topics for every document m accordingly. A representation of the individual topics is usually given by a list of n words having highest probability in a topic. This is done by sorting the individual $\phi_{\cdot,k}$ in descending order and retrieving the first n entries afterwards as shown in Table 1.

Matching LDA model posterior distributions

The target is to define a function $\text{sim}(p(\mathbf{w}|z_k), p(\mathbf{w}|z_k^*))$ that allows a satisfying separation of topics, so that we are able to define a threshold of similarity that adequately matches identical topics across different models. The outcome of the similarity function $\text{sim}(\cdot, \cdot)$ should span a wide range of values, i.e. the function's outcome for similar topics and dissimilar topics has to differ significantly. Otherwise, setting a general optimal threshold obviously becomes practically impossible. The posterior distributions over words given the topics $\phi_{\cdot,k} = p(\mathbf{w}|z_k), (k = 1, \dots, K)$ can be interpreted as the semantic context or latent structure of the analyzed text corpus. These distributions are used to summarize the corpus contents as short lists of words, giving the intuition that there is only a small number of terms that form the main context of a topic within an LDA model. In Figure 1 we show that indeed only a minor count of terms represent a major probability portion within a topic $p(\mathbf{w}|z_k)$. To demonstrate this property, we built the cumulative distribution function (CDF) for an example

topic after sorting the distribution’s probability values in descending order. Although the distribution over words for a topic depends on the β prior of the model, we observed this behavior in models where the inferred topics allow an intuitive interpretation (Chang, Boyd-Graber, Gerrish, Wang, & Blei, 2009; Newman, Lau, Grieser, & Baldwin, 2010). Another reading of this finding is the fact that only high probability words are of importance for the topic since all words (belonging to a long tail) of low probability in a topic have about the same mass within all other topics. Thus, the probability mass of the words with highest probability is also constant across topics and independent of the actual words. Considering this and the topic representations in Table 1, the intuition arises, that a similarity function based on simple word matching in sublists of high probability terms from the posterior distributions $p(\mathbf{w}|z_k)$ and $p(\mathbf{w}|z_k^*)$ of different models can help considerably to track topics across different models.

Related Work

To distinguish our work we will briefly discuss related approaches in more detail. The ability of topic models to analyze changes in semantic contexts of continuous document streams was introduced by (D. Blei & Lafferty, 2006; AlSumait et al., 2008). As already described in section 1, these approaches use the outcome of a model from a previous chunk of data, e.g. a time slice $t - 1$, and utilize it as the prior for a new succeeding time slice t . In both setups the authors use a fixed number of topics to be inferred from the data⁴. In detail, they use the posterior distribution $p(\mathbf{w}|z_k)$ of topic k to formulate a prior β_k for model inference in succeeding data chunks. In a setup dealing with continuous streams or consecutive corpora, the main idea is, that contents in a data stream are stable over a certain time frame. Although, the method of generating priors from posteriors differs in both approaches, the idea of keeping the context of the corpus over time is the same. To incorporate knowledge about changes or stability of topics, measures like the KL divergence are used (see (AlSumait et al., 2008)). Finally, the change of a topic’s context is anticipated when the topic’s distribution in previous models differs from the current one.

Based on these ideas, analysis of the topics’ evolution in a corpus is feasible by fixing the number of topics and dividing the data into chunks or time slices. Unfortunately, this approach is limited to using the same number of topics in each chunk, which is not optimal when the number of concepts in a text stream e.g. in news streams changes. In that case, having a fixed number of topics is inapt. Consequently, optimization of the topic models for each time slice/chunk of data separately seemed desirable to us, especially in the setting of highly dynamic news data streams. Thus independent topic models for each time slice have been used in our approach. Optimization includes inference of hyperparam-

⁴Both approaches rely on LDA as the topic-generating statistical model, and thus are bound to define the LDA model parameter K , i.e. the number of topics the model produces

eters⁵ and determining an optimal number of topics for the data (as in (Griffiths & Steyvers, 2004)). We produce the relationship between models afterwards via the proposed approach. The benefit of this idea is that we can detect newly arising as well as vanishing topics with exact quantities and can distributively process the models on different CPU’s or machines.

Similarity measures

Different measures exist for comparing probability distributions (or real valued vectors in \mathbb{R}^V as a generalization thereof). Since we are working with different corpora or text chunks of unequal size we cannot use absolute word counts to deduce the probability distributions $p(\mathbf{w}|\mathbf{z})$ for each model as has been done by (AlSumait et al., 2008). Instead, we use normalized probability distributions over the vocabulary as a representation of topics that are given by $\phi_{\cdot,k}$ for each topic k . Naturally, elements of $\phi_{\cdot,k}$ are probabilities in the range $]0, 1[$. Thus using metrics based on point distances in euclidean space will result in very low values in general that tend to be useless to correctly distinguish between a match or a mismatch.

In our experiments we will create similarity matrices, hence we defined the proposed measures as similarities. The following measures have been evaluated in our experiments: *Jensen-Shannon divergence (JSD)*: Since we are dealing with probability distributions we chose this measure as a smoothed and symmetric alternative to the Kullback-Leibler (KL) divergence, which is a standard measure for comparing distributions. Note that the outcomes of JSD need to be normalized. The normalized values can then be transformed into a similarity measure by subtracting them from 1. In the following equation we set the distributions $p(\mathbf{w}|z_k)$ and $p(\mathbf{w}|z_k^*)$ to be compared as P and Q and use:

$$JSD(P||Q) = \frac{1}{2}D(P||M) + \frac{1}{2}D(Q||M) \quad (2)$$

where

$$M = \frac{1}{2}(P + Q). \quad (3)$$

Cosine similarity: Interpreting the posterior distributions $p(\mathbf{w}|\mathbf{z})$ for a topic model as weighted word vectors, the cosine similarity is an unorthodox but nevertheless valid measure. Since it describes the angle between two vectors, the similarity is independent of the norm of the vectors and gives equal results as for unnormalized word counts. Note that the cosine similarity almost identical to the normalized correlation coefficient (Manning & Schütze, 1999) in our case: Since, due to the low probability of most words, the word distribution’s mean is close to 0, the calculation of the correlation between

⁵Hyperparameters strongly influence the model outcome and thus must be optimized according to the intended task. One might analyze newspapers based on editorial departments, whereas others might search for very atomic topics. The latter, however, will not be possible using the mentioned prior based approaches due to a high variance in the topic counts.

two probability vectors will result in a value very close to the normalized correlation coefficient and won't take any negative values. For that reason computation of the correlation between two vectors has been skipped for its redundancy. We set the distributions $p(\mathbf{w}|z_k)$ and $p(\mathbf{w}|z_k^*)$ to be compared as A and B and use:

$$s(A, B) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (4)$$

Dice's coefficient: Consider the summary of topics by a list of the top n words per topic as in Table 1. Looking at the lists, e.g. the japan topics, we can identify the similarity or the overlapping of the contents by just inspecting the words without using their actual probability. Following this idea, we also consider another similarity measure based on word sets, Dice's coefficient, that might seem unusual to compare different probability distributions. We set the words from the sorted distributions $p(\mathbf{w}|z_k)$ and $p(\mathbf{w}|z_k^*)$ to be compared as X and Y .

$$s = \frac{2|X \cap Y|}{|X| + |Y|} \quad (5)$$

Experiments

Our dataset consists of 2,133 news articles from five consecutive days (March 10th through 15th 2011) containing 64,674 unique word types, obtained through the API of the British newspaper *The Guardian*. Within this period there are two dominating news topics that we use as a basis for our experiments. Those are the riots in Libya and the consequences of the earthquake and tsunami catastrophe in Japan. Furthermore we will use one topic consisting of only stop words as a negative example with respect to the Japan and Libya topics, to test the performance of the similarity measures. In order to evaluate the different similarity measures we fit different topic models (one for each day) with comparable results. Since we also created a single corpus for each of the consecutive days we handpicked topics from the models. These topics are illustrated in Table 1 where we sorted the words by their probability and chose the 20 most probable words to summarize the contexts.

Based on these hand selected topics we built similarity matrices comparing the similarities of all topics using the similarity measures described in section 5. Additionally we tested the similarity measures on different word subsets of the topics. This means that we set all the probabilities within a topic distribution $p(\mathbf{w}|z_k)$ to 0 except those for the most probable n words. In our setup we chose $n \in \{2, 5, 10, 20, 40, 80, 160, 320\}$. From the intuition that the most probable words sufficiently define a topic's context, we expect a more unambiguous and robust similarity matrix for comparisons based on small n . To decide how robust the similarities are, we measure the absolute deviation between the true and the desired similarity for each entry in a similarity

matrix for a specific word sub-set. We average this value over all similarities for each topic. The mean absolute deviation for this setting is defined as

$$MD = \frac{1}{N_{topics}^2} \sum_{i=1}^{N_{topics}} \sum_{j=1}^{N_{topics}} \|s_{ij} - s_{ij}^*\| \quad (6)$$

where N_{topics} is the number of topics included in the similarity matrix, s_{ij} is the measured similarity and s_{ij}^* is the desired similarity. If two topics match, the desired similarity s_{ij}^* is equal to 1 whereas in contrast to that, the desired similarity for non-matching topics will be set to 0. If the intuition that the n most probable words sufficiently define the topic context/meaning is correct, incorporating only semantically relevant words into the comparison results in a decrease of the mean absolute deviation. To measure this behavior we calculate the mean absolute deviation of all elements within a similarity matrix for all defined values of n . To select the optimal similarity measure in combination with the optimal sub-set of words, we will determine the combination for which the mean absolute deviation has a minimum.

Note that the selection of the optimal sub-set of words needs to be rechecked for new tasks in new text sources since the probability distributions, and thus the number of meaningful words of the topics, strongly depend on those preferences.

Results

Performing the experiments with the procedure described above gives 27 similarity matrices.⁶ For each matrix we calculated the mean absolute deviation of its entries. Figure 2 shows the performance of the different similarity measures. The x-axis represents the number of the most probable words used whereas the y-axis corresponds to the mean absolute deviation. Cosine similarity quite surprisingly yields the best results, i.e. the lowest mean absolute deviation for a sub-set of 10-40 words. A minimum of the mean absolute deviation of similarity values means that we have a higher tolerance to set a threshold. Similarities are close to their desired values and similarity values of positive and negative matches are spread over a wider range. Also, the intuition is verified that the spreading between the similarity values, and hence the distinguishability, rises when we exclude words from the comparison that scatter their probability mass over a large number of other topics: Incorporating all words of a topic's word distribution into the comparison always results in a certain amount of similarity among topics in a corpus. This is caused by the fact, that many words (belonging to the long tail of low probability words in a topic's word distribution) spread their probability mass across all topics in the corpus, i.e. they belong to the long tail of all other topics as well. Obviously, this provokes similarity to some degree, even if topics are not related at all. Thus, taking away low probability words results in higher similarity of topics that effectively mean the same.

⁶We compare three similarity measures. For each measure we built nine different similarity matrices based on the comparison of the topics with only the top n words left.

Table 1: Selected Topics from consecutive days 10th -15th March 2011.

Date	Shortname	Top 20 Words
12-03-2011	japan1	Japan nuclear plant tsunami earthquake reactor power Japanese disaster radiation water damage quake plants country Tokyo explosion reactors Fukushima reports
13-03-2011	japan2	nuclear Japan tsunami power earthquake reactor Japanese water disaster plant radiation crisis plants magnitude fuel reactors aftershocks rescue Friday prefecture explosion
14-03-2011	japan3	nuclear Japan reactor power plant Japanese earthquake tsunami explosion disaster Tokyo rescue reactors energy plants crisis radiation JST safety water
15-03-2011	japan4	nuclear Japan plant power radiation Japanese reactor reactors fuel earthquake levels Tokyo water disaster tsunami fire level crisis agency safety
10-03-2011	libya1	Libya Gaddafi forces military zone no-fly Nato Libyan Libyan oil foreign rebels rebel council Ras.Lanuf France fighting regime defence country
12-03-2011	libya2	Gaddafi Benghazi MP country regime revolution revolutionary Libya forces GG international council countries intervention foreign eurozone Libyan no-fly city army
13-03-2011	libya3	Gaddafi Libya oil foreign Arab Europe intervention no-fly Iraq zone support military forces regime rebels security western uprising Egypt Tunisia
14-03-2011	libya4	Cameron Labour Libya zone Gaddafi no-fly Miliband Balls Britain vote tax campaign action plan party Clegg ministers Labour rebels referendum
15-03-2011	libya5	no-fly zone Bahrain forces Gaddafi military Libya troops security rebels foreign torture regime Benghazi told Saudi.Arabia Britain France G8 town
15-03-2011	stopwords1	years public make work pay world made good UK back part long ve don day Germany week big report

As we stated before, these properties can vary for different text sources and tasks. Since other models need to fulfill different requirements for other content analysis tasks, they are often run with different sets of parameters or other preconditions. Hence, the proposed procedure needs to be reproduced for other text sources and/or models in order to select the optimal size of word sub-sets. However, cosine similarity definitely yields best results in the context of our matching process.

Applications and Future Work

In this paper, we presented a method to match the outcome of different topic models on the basis of the word distributions $p(w|z)$. With this setting it is possible to train topic models on little chunks of text data and match the outcomes afterwards. An application for this is the generation of a topic models per hour, day, month or year where we can match the outcomes easily. With this on hand we can track and detect topics within diachronic news, patent or social media text sources. Furthermore we can handle very large datasets by dividing the text sources into document sub-sets and distributing the model training to many machines. Afterwards we can

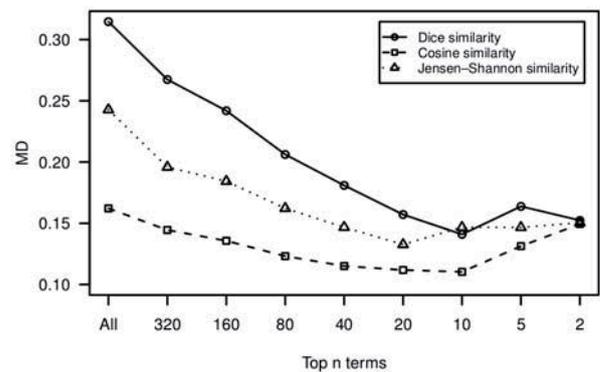


Figure 2: Mean absolute deviation for sub-sets of words

match the outcomes and give an accumulated view onto the whole corpus.

Future work will be focused on the selection of a threshold for different text sources and the definition of word sub-sets to use. Because of the diverse properties of certain text sources, specifying a general threshold for matching the topics proved to be inappropriate. For every text source, precision and recall of topic matching have to be optimized separately. To address this we will establish a procedure to test specific text sources for an optimal threshold. In (Silva, Stasiu, Orenco, & Heuser, 2007) a promising approach is shown, which can be adopted to this problem. Using this work it is possible to address the topic tracking problem with a mixture of lightweight similarity measures and simple fast processable topic models. With the connection of similar topics, time series data of consecutive chunks of text data e.g. consecutive days can be built, which can then be further analyzed to detect trends, unusual behavior or seasonal effects.

References

Allan, J. (2002). Introduction to topic detection and tracking. In J. Allan & W. B. Croft (Eds.), *Topic detection and tracking* (Vol. 12, p. 1-16). Springer US.

AlSumait, L., Barbará, D., & Domeniconi, C. (2008). Online lda: adaptive topic models for mining text streams with applications to topic detection and tracking. *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, 3–12.

Bishop, C. M. (1999). Latent variable models. In M. I. Jordan (Ed.), *Learning in graphical models* (pp. 371–403). MIT Press.

Blei, D., & Lafferty, J. (2006). Dynamic topic models. In *Proceedings of the 23rd international conference on machine learning* (pp. 113–120).

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3, 993–1022.

Chang, J., Boyd-Graber, J., Gerrish, S., Wang, C., & Blei, D. (2009). Reading tea leaves: How humans interpret topic models. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, & A. Culotta (Eds.), *Advances in neural information processing systems 22* (pp. 288–296).

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl 1), 5228–5235.

Manning, C., & Schütze, H. (1999). *Foundations of statistical natural language processing*. Cambridge Mass.: MIT Press.

Newman, D., Lau, J. H., Grieser, K., & Baldwin, T. (2010, June). Automatic evaluation of topic coherence. In *Human language technologies: The 2010 annual conference of the north american chapter of the acl* (pp. 100–108). Los Angeles, California: Association for Computational Linguistics.

Silva, R. da, Stasiu, R. K., Orenco, V. M., & Heuser, C. A.

(2007). Measuring quality of similarity functions in approximate data matching. *J. Informetrics*, 1(1), 35–46.

Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M. (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association*.

Wang, X., & McCallum, A. (2006). Topics over time: a non-markov continuous-time model of topical trends. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 424–433.

Appendix: Example similarity matrices

Figures 3 and 4 show the difference between an unassertive and a confident similarity matrix. A similarity of one corresponds to white, zero similarity is drawn in black. Note that we have a small amount of similarity between all topic pairings if we include all words for a match.

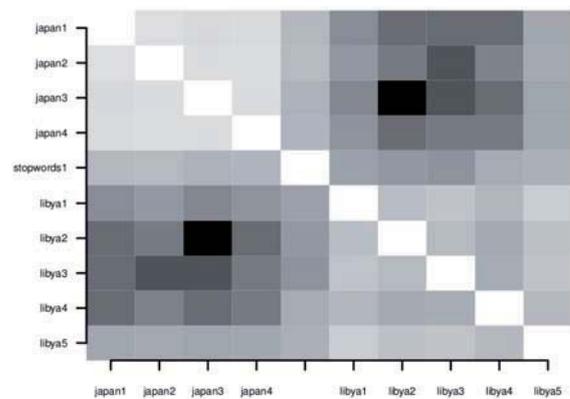


Figure 3: Similarity matrix with all words based on Jensen-Shannon divergence. Not only matching topics exhibit similarity.

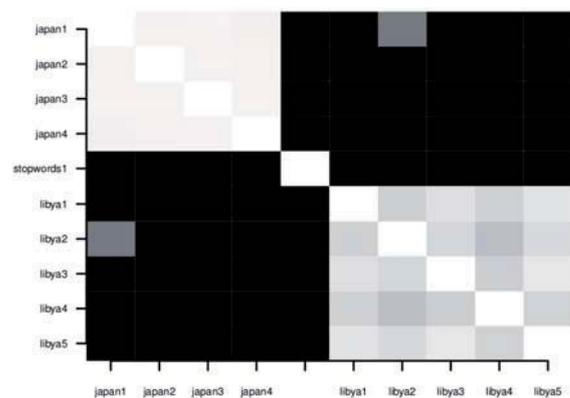


Figure 4: Similarity matrix with matching of 20 most probable words based on cosine similarity. High similarity is given to matching topics only.

A Self-Organized Neuronal Comparator

Guillermo A. Ludueña (ludueña@itp.uni-frankfurt.de)

Institute for Theoretical Physics, Max-von-Laue-str. 1,
60438 Frankfurt am Main, Hessen, Germany

Claudius Gros (gros07@itp.uni-frankfurt.de)

Institute for Theoretical Physics, Max-von-Laue-str. 1,
60438 Frankfurt am Main, Hessen, Germany

Keywords: Unsupervised Learning; Neural networks; Self-organization in complex systems; Artificial intelligence

Abstract

In order to develop a complex targeted behavior, an autonomous agent must be able to relate and compare the information received from the environment and internally generated (Billing, 2010). For example it is often necessary to decide whether the visual image currently being perceived is a similar image encoded in some form in memory.

Neural learning architectures hence need a unit, a comparator, able to compare several inputs encoding either internal or external information, like predictions and sensory readings. Without the possibility of comparing the values of prediction to actual sensory inputs reward evaluation and supervised learning would not be possible.

Comparators are usually not implemented explicitly, necessary comparisons are commonly performed by directly comparing one-to-one the respective activities, see for instance (Bovet & Pfeiffer, 2005a, 2005b). This implies that the characteristics of the two input streams (like size and encoding) must be provided at the time of designing the system.

It is however plausible that biological comparators emerge from self-organizing, genetically encoded principles, which allow the system to adapt to the changes in the input and in the organism.

We propose an unsupervised neural circuitry, where the function of input comparison emerge via self-organization only from the interaction of the system with the respective inputs, without external influence or supervision.

The proposed neural comparator neural circuit adapts according to the correlations in the information streams received as inputs. The system consists of a multilayer feed-forward neural network which follows a local output minimization (anti-Hebbian) rule for adaptation of the synaptic weights.

The local output minimization allows the circuit to autonomously acquire the capability of comparing the neural activities received from different neural populations, which may differ in the size of the population and in the neural encoding used.

References

Billing, E. A. (2010). Cognitive perspectives on robot behavior. In *Proceedings of the 2nd international confer-*

Density of (output, distance) pairs, log scale

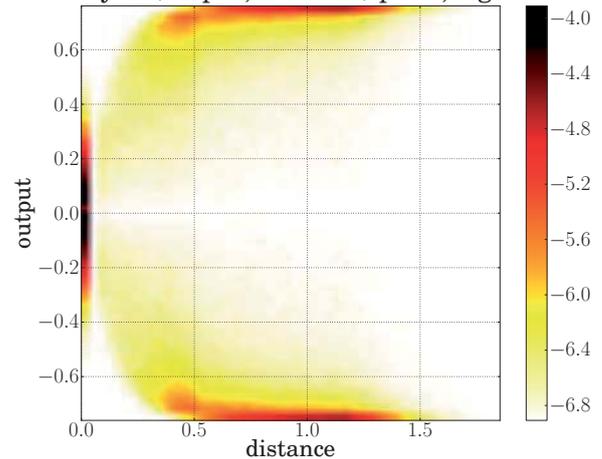


Figure 1: Density of output-distance pairs from the last 10% of 700000 iterations of the comparator, comparing inputs of dimension $N = 5$ under the same encoding. 20% of the Input were randomly-chosen equal vectors. The network self-organizes to assign large output when the vectors are separated by a large euclidean distance and small output when the distance is zero. Thus the output behaves as a sigmoid of the distance between input vectors, despite the network having no information about the input being two vectors or about how to calculate an euclidean distance.

ence on agents and artificial intelligence: Special session on computing languages with multi-agent systems and bio-inspired devices (p. 373-382).

Bovet, S., & Pfeiffer, R. (2005a). Emergence of coherent behaviors from homogenous sensorimotor coupling. In *Proceedings of the 12th international conference on advanced robotics (icar)*.

Bovet, S., & Pfeiffer, R. (2005b). Emergence of delayed reward learning from sensorimotor coordination. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (iros)*.

Symposium on Human Performance Modeling

Wayne D. Gray (Organizer)
Rensselaer Polytechnic Institute

David Kaber and Guk-Ho Gil and Sang-Hwan Kim
North Carolina State University

Naom Ben-Asher and Joachim Meyer
Ben-Gurion University

Shi Cao and Yili Liu
University of Michigan

Cleotilde Gonzalez (Panel Member)
Carnegie Mellon University

Glenn Gunzelmann and Kevin Gluck (Panel Members)
U. S. Air Force Research Laboratory

This symposium is co-sponsored by the Human Performance Modeling Technical Group (HPM-TG) of the Human Factors & Ergonomics Society. Three Research Talks and a Panel Discussion were presented. Each talk used a different style of cognitive modeling and addressed a different problem of interest to the human factors community. For the Panel Discussion, three additional members of the HPM-TG joined with our three speakers in a round table discussion of the similarities and differences between cognitive modeling in applied versus basic science.

Keywords: human performance modeling, human factors

Introduction

The Human Performance Modeling Technical Group (HPM-TG) of the Human Factors and Ergonomics Society (HFES) is proud to sponsor its first Symposium at ICCM. Although much of the work presented at ICCM focuses on basic research, it is clear that ICCM recognizes the value of applied cognitive modeling. Indeed, a long tradition at ICCM is the *Siegel-Wolf Award for Best Applied Paper*. Not only does this reward reflect the value of applied modeling, but it is named after two men who were HFES members in the early days of human performance modeling.

The spirit of Siegel and Wolf lives on in the HFES and ICCM communities even though both largely go their separate ways. This symposium is intended as the first in a long term exchange that we hope will enrich ICCM and the HPM-TG. Those desiring a snapshot of the recent history and current status of cognitive modeling in human factors should see Gray (2008a, 2008b). Those interested in details from the early days of human performance modeling should see two excellent papers by Pew (2008, 2007).

Technical Talks

An Accessible Cognitive Modeling Tool for Evaluation of Pilot-Automation Interaction - Kaber, Gil, & Kim

One of the main limitations of existing approaches to complex human-in-the-loop system design is the requirement for empirical data as a basis for alternative design selection. Experimental studies can be time consuming and costly. In addition, design decisions are often based on collections of de-

sign guidelines with limited theoretical explanations for why such guidelines may be effective from a human information processing (HIP) perspective. The lack of a cognitive explanation limits understanding of when and how guidelines can be applied. In order to better support conceptual design, various cognitive modeling techniques and tools have been developed based on HIP architectures. However, these techniques and tools also have several limitations from a design perspective. Existing tools are not easy to use and designers or developers may need extensive training and practice in use. Furthermore, there is currently no fundamental set of tool capabilities, such as providing a task workload analysis or identifying patterns of HIP (e.g., memory use), simulating visual object use (e.g., eye movements), providing interface design support, etc. This research integrated various capabilities of existing modeling tools into a new enhanced cognitive modeling language based on GOMS (Goals, Operators, Methods, and Selection Rules).

While GOMS modeling methods and the GOMS language are considered easy to learn and use, the modeling approach has several limitations. The language is limited to representing expert behavior in tasks. In addition, GOMS models do not support modeling of lower-level behaviors, such as specific forms of visual processing (e.g., foveal vs. peripheral) as well as parallel processing of visual and motor operations. Another major limitation of GOMS modeling is that the operator time estimates are deterministic. Therefore, model output may not accurately represent individual differences in performance or the stochastic nature of human behavior in complex tasks. On the basis of these limitations, this re-

search developed a new computational cognitive modeling tool using an enhanced-GOMS language to aid complex system designers in assessing human performance and errors in using complex automated systems.

A human information processing model described by Wickens (1992) was used in this research as a cognitive architecture to support and constrain E-GOMSL model coding. New operators as part of the E-GOMS language were defined with four properties based on Wickens HIP model, including: (1) the cognitive processing channel used, (2) control objects, (3) operator syntax and (4) operator times. Each channel has control objects (e.g., flow of control, parallel processing, etc.). E-GOMSL operator syntax is similar to GOMSL and NGOMSL operator syntax. The new operator set was primarily based on NGOMSL operators, as originally defined by Kieras (1997); however, the control structure of EGOMSL models follows GOMSL models, in order to support compilation and model execution with a simulation engine. As the second step in E-GOMSL development, stochastic variables were defined to represent operator times in behavior models. Because computational cognitive modeling is conceptually similar to discrete event simulation of human task performance, methods used in systems simulation for representing or quantifying event processing times have been extended to cognitive modeling. An overall stochastic time estimate can be calculated as the summation of all operator time estimates in an E-GOMSL model. The time estimates can be considered to represent the range of human performance, including normal (average), super skill and slacker behavior (Niebel & Freivalds, 2003).

The cognitive modeling tool development included: a prototyping module; a user activity flow diagram (AFD) development module; an AFD to E-GOMS language translator; an E-GOMSL editor; a model parser and compiler; and a model simulation tool and report generator. A designer is able to use images to define a prototype including visual and non-visual objects (e.g., auditory interfaces). The designer can also develop an AFD based on the results of a cognitive task analysis (CTA) involving expert operators. The AFD is directly translated to E-GOMS by the translator module. After coding the model, the parser and compiler can be used to obtain a quantitative analysis including task execution times based on stochastic estimates of individual operation times and a workload analysis. With these results and the GOMSL models, the simulator can be used to visualize the flow of HIP, represent patterns in HIP, and present a graphical workload analysis. Last, the report generator can be used to produce a summary of the quantitative analysis and simulation.

In order to validate the results of the modeling tool, a flight simulator experiment was conducted with a futuristic form of cockpit automation (a Continuous Descent Approach (CDA) tool for flight route replanning). A CTA was conducted to identify pilot behaviors and to generate a data set for vali-

ation of the cognitive model output. An E-GOMSL model of pilot behavior with the CDA tool was compared against the experiment data. There was a marginal positive correlation between the model and pilot experiment task times ($p = 0.3489$, $n = 27$, $p = 0.0745$). Comparison of E-GOMSL model outputs at various points in task performance with actual pilot heart rate responses (correlation analysis: $p = 0.2055$, $p = 0.0181$) indicated working memory (WM) item counts from a model could serve as a basis for predicting automation and task-induced cognitive load. In general, when the model predicted WM count was at a minimum, the HR response for pilots revealed low arousal. When the model-predicted count was at a maximum, the HR response for pilots revealed high arousal. These findings indicate that the E-GOMSL model may explain differences in automation or task-induced cognitive load in terms of WM use.

In line with expectations, results demonstrated the modeling approach to support accurate explanation and prediction of human behaviors and performance in using complex systems. The findings of this research support the new EGOMSL tool use during the conceptual design of complex human-in-the-loop systems and/or interfaces.

Modeling Users' Risk-related Behaviors when Interacting with Computer Systems - Ben-Asher & Meyer

Computer security is gaining importance because of the ubiquitous introduction of computers into all domains of life, the use of computers to store and access sensitive information of various kinds (e.g., bank accounts, medical records), and the increasing use of mobile devices to access these systems. In recent years, it has become clear that the human user is often the weakest link in computer security. Even if the system requires long and complex passwords, it becomes unsecured if users paste them on their computer monitors. Also, even if one has sophisticated algorithms for detecting malicious software, for instance on websites, the user may override the system recommendation and become exposed to these threats. The design of adequate computer security requires us therefore to predict the user's risk-related behaviors with computer systems. An adequate understanding of user behavior and the prediction of user actions will allow us to design systems and security measures, so that users will tend to act securely.

Two main issues need to be considered when modeling users' risk-related behavior with computer systems. First, very little behavioral data are available on how users cope with security risks. The main reason is that publishing information on how users, for instance, respond to indications of security threats and what affects their responses to these threats can possibly be exploited by those who generate threats and increase the severity of threats. Second, the user's risk-related behavior may actually be a combination of several different, interrelated behaviors. We suggest the

notion of a "triad of risk-related behavior" (Ben-Asher & Meyer, submitted for publication), where the user's coping with security issues in computer systems is affected by the user's exposure to risk, the installation and setting of security features, and the response to risk-related communications.

We developed an experimental system, based on the Tetris game, to allow us to collect empirical data on all three behaviors. In our version of the game users try to accumulate as many points as possible but, different from the usual Tetris, completed rows remain on the screen until the user decides to "save them" (an action that stops the game and is therefore costly users are paid according to their performance, and the game is limited in time). Occasionally "attacks" occur in which a malicious virus deletes part of the cells the user has accumulated and hasn't saved, yet. The user sees alerts from a security system (with imperfect validity) about the possibility of an attack.

To apply the insights gained from the experiments for the generation of design recommendations, it is important to model the different behaviors and their interactions. One model, based on the Memo-workbench, focuses on the analysis of the user-system interaction (Möller, Ben-Asher, Englert, & Meyer, 2011) We have begun to develop models, adopting three additional modeling approaches:

1. A cost-benefit model to predict the optimal user actions, given the properties of the system.
2. A reinforcement-based learning model in which we attempt to predict the changes in the security system settings and in the users' tendency to expose themselves to risk and to respond to alerts.
3. A system dynamics model that analyzes the feedback loops in the process.

All models start with the parameters of the experiment for a given condition and then generate predictions of user behavior, which we compare to empirical results. We discuss the challenges that exist when trying to model a complex behavior in an experimental microworld in which users' actions result from the combination of different, interrelated behaviors. We also discuss the advantages and problems with each of the different modeling methodologies we employed and point towards the requirements for a comprehensive modeling of users' risk-related behaviors with computer systems.

ACTR-QN: Integrating Queuing Network and ACT-R Cognitive Architectures - Cao & Liu

ACTR-QN is a cognitive architecture that integrates Adaptive Control of Thought-Rational (ACT-R) and Queuing Network (QN) architectures. ACT-R (Anderson et al., 2004) has sophisticated declarative memory mechanism based on chunk activation and procedural memory mechanism based on production rule utility. It is particularly powerful in modeling cognitive tasks such as learning and problem solving. QN (Liu, Feyen, & Tsimhoni, 2006), on the

other hand, has its mathematical basis of queuing theory, which supports the modeling of complex mental structures and scheduling mechanisms. As a result, the QN architecture has its strength in modeling multitask performance and mental workload. The integrated ACTR-QN represents ACT-R as a QN, whose servers are ACT-R modules and buffers with information paths in between and entities correspond to ACT-Rs information units such as chunks and production rules. Theoretically, ACTR-QN allows modelers to combine the power of ACT-R and QN and examine a wider range of fundamental cognitive issues from new perspectives, for example, modeling multitask performance involving complex cognitive tasks.

For cognitive engineering applications, a software program implementing ACTR-QN has been developed using Micro Saint Sharp (www.maad.com), which was chosen because it provides natural supports for QN modeling and visualization. Further, it is the same platform on which IMPRINT is implemented. Full integration of ACTR-QN was achieved by porting ACT-R (v 6.0) from Lisp into Micro Saint Sharp (C#). In addition, ACTR-QN also integrated the PG-C version of utility computation and the recent work on threaded cognition. Workload modeling capability was inherited from QN using server utilization.

Each model in ACTR-QN has two parts: the mind and the task. To build the mind part, including chunks, production rules, and parameters, ACTR-QN reads and uses the same ACT-R codes. For the task part (displays and controls), easy-to-use templates have been developed to model both static tasks, in which display stimuli are predetermined and not affected by responses, and dynamic tasks, in which responses affect display stimuli dynamically, such as driving. Modelers simply need to follow instructions and specify the parameters of a task, such as the frequency of a tone or the geometry of a road.

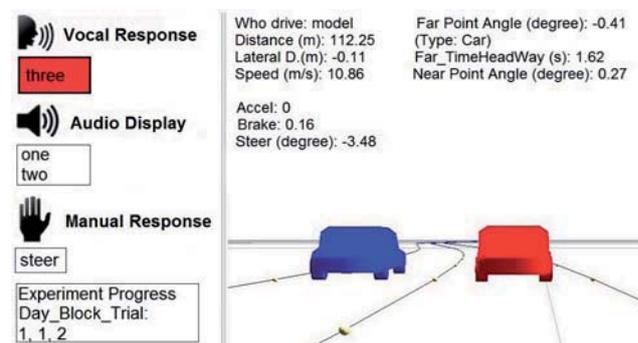


Figure 1. Task visualization in ACTR-QN

ACTR-QN provides visualization of the mind, the task, and mental workload. Figure 1 illustrates the visualization of model performing a dual task of auditory-vocal arithmetic

addition (left) and driving (right). Model testing shows that ACTR-QN produces the same results as ACT-R for typical cognitive tasks. Future research will examine the benefits of further integration between ACT-R and QN cognitive architectures, especially in modeling performance and workload in multitask scenarios involving complex cognition.

Panel

During the Conference, part of the Symposium presentation included a discussion among the three panelists, three presenters, and the audience regarding the differences and similarities of cognitive modeling for human factors applications versus other types of cognitive modeling. It is unfortunate that a transcript of this exchange cannot be provided here.

Acknowledgements

Kaber, Gil, and Kim's work was supported by NASA Ames Research Center under Grant No. NNH06ZNH001. Mike Feary was the technical monitor. The opinions and conclusions expressed in the associated paper are those of the authors and do not necessarily reflect the views of NASA. This research was completed while the third author, Sang-Hwan Kim, worked as research assistant at North Carolina State University.

Gray's work was supported, in part, by grant N000141010019 from the Office of Naval Research, Dr. Ray Perez, Project Officer.

References

- Anderson, J. R., Bothell, D., Byrne, M. D., Douglas, S., Lebiere, C., & Quin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036–1060.
- Ben-Asher, N., & Meyer, J. (submitted for publication). The triad of risk-related behavior in computer security: effects of skill, costs and attack likelihood.
- Gray, W. D. (2008a). Cognitive architectures: choreographing the dance of mental operations with the task environments. *Human Factors*, *50*(3), 497–505.
- Gray, W. D. (2008b). Cognitive modeling for cognitive engineering. In R. Sun (Ed.), *The cambridge handbook of computational psychology* (pp. 565–588). New York: Cambridge University Press.
- Kieras, D. E. (1997). A guide to GOMS model usability evaluation using NGOMSL. In M. Helander, T. K. Landauer & P. Prabhu (Eds.), *Handbook of human-computer interaction* (2nd Edition, pp. 733–786). Elsevier.
- Liu, Y., Feyen, R., & Tsimhoni, O. (2006). Queuing network-model human processor (qn-mnp): a computational architecture for multitask performance in human-machine systems. *ACM Transactions on Computer-Human Interaction*, *13*(1), 37–70.
- Möller, S., Ben-Asher, N., Englert, K.-P., & Meyer, J. (2011). Modeling the behavior of users who are confronted with security mechanisms. *Computers & Security*, *30*(4), 242–256.
- Niebel, B. W., & Freivalds, A. (2003). *Methods, standards, and work design* (12th Edition). Boston: WCB/McGraw-Hill.
- Pew, R. W. (2008). More than 50 years of history and accomplishments in human performance model development. *Human Factors*, *50*(3), 489–496.
- Pew, R. W. (2007). Some history of human performance modeling. In W. D. Gray (Ed.), *Integrated models of cognitive systems* (pp. 29–44). New York: Oxford University Press.
- Wickens, C. D. (1992). *Engineering psychology and human performance* (Second). New York: HarperCollins.

Author Index

Young-Suk Ahn	88	Uwe Drewitz	265
Olurotimi Richard Akinlofa	129	Varun Dutt	88, 163
Eduardo Alonso	234	Cornelia Eckers	289
Eef Ameel	167	Alonso Eduardo	96
Andreas Ammann	144	Eyad Elyan	129
John Anderson	249	Bruno Emond	43
Krzysztof Andrelczyk	181	Klaus-Peter Engelbrecht	80
Nina Arnold	137	Stella Faerber	13
Joscha Bach	271	Michael Fairbank	234
Jerry Ball	131	Thomas Fangmeier	19
Alexandra Barchunova	74	Johannes Folger	13
Ute Bayen	137	Birte Forstmann	273
Philip Beaman	25	Stan Franklin	283
Michael Beckmann	127	Matthias Frorath	19
Thierry Bellet	59	Wai-Tat Fu	103
Noam Ben-Asher	88, 163, 324	Joachim Funke	303
Stefano Bennati	157	Guk-Ho Gil	324
Udo Boehm	273	Kevin Gluck	240, 324
Jean-Charles Bornard	59	Stefan Gluege	123
Joscha Borné	303	Cleotilde Gonzalez	88, 163, 324
Jelmer Borst	249	Linn Gralla	230
Daniel Brand	297	Jonathan Gratch	55
Stefan Brandenburg	265	Wayne Gray	66, 324
Duncan Brumby	140	Claudius Gros	323
Sven Brüssow	19	Enrico Grosso	139
Martin Butz	187	Dominique Gruyer	59
Shi Cao	217, 324	Olivia Guest	259
Claus-Christian Carbon	13	Glenn Gunzelmann	224, 324
Xiuli Chen	193	Kai Hamburger	82
Adam Chuderski	181	Robert Hamilton	311
Bernard Claverie	59	J. Isaiah Harbison	169
Gilles Comeau	43	Kerstin Haring	275
Rick Cooper	259	Kim Hartmann	123
Marissa Daigle	224	Robert Haschke	74
Eddy Davelaar	31, 169	Petra Hendriks	101
Alberto De Obeso Orendain	49	Oliver Herbort	187
Stephane Deline	281	Jeremiah Hiam	117
Barbara Deml	123	Daniel Holt	303
Nate Derbinsky	109, 205	Patrik O'Brian Holt	129
Michael Dougherty	169	Andrew Howes	140, 193

Author Index

Erika Hussey	169	Larry Moore	224
Kenryo Indo	232	Jonathan Morgan	117
Christian Janssen	140	Jan Moringen	74
Phil Johnson-Laird	151	Hendrik Neumann	123
Jacques Juhel	281	Christiane Neuschaefer-Rube	289
Ion Juvina	55	Andreas Niekler	317
Patrick Jähnichen	317	Menno Nijboer	241
David Kaber	324	Enkhbold Nyamsuren	211
Ioanna Katidioti	315	Alessandro Oltramari	55
Ryan Kaulakis	117	Jean-Christophe Paris	59
William Kennedy	1, 55	Robert Patterson	1
Sangeet Khemlani	151	David Peebles	37
Sang-Hwan Kim	324	Georgi Petkov	7
Muneo Kitajima	291	Peter Pirolli	103
Markus Knauff	82	Mondragon Plaza	96
Boicho Kokinov	7	Athanassios Protopapas	315
Lars Konieczny	275	Felix Putze	303
Jonathan Kottlors	297	Gloria Pöhler	127
Michael Kotzyba	123	Marius Raab	13
Bernd Kroeger	289	Marco Ragni	19, 157, 275, 297
Antje Krumnack	82	Ralf Reichmuth	255
Johan Kwisthout	199	Rainer Reisenzein	313
John Laird	109, 205	David Reitter	253
Nicholas Lange	31	Frank Ritter	55, 117
Christian Lebiere	55	Helge Ritter	74
David Pierre Leibovitz	72	Robert Robere	111
Leif Kåre Lende	175	Paul Rosenbloom	305
Q. Vera Liao	103	Florian Röser	82
Yili Liu	217, 324	Matthias Rötting	255
Johannes Lohmann	187	Joseph Sanford	117
Ludovica Lorusso	139	Stefan Schaffer	253
Max Lotstein	151	Simone Schineller	13
Guillermo Luduena	323	Ute Schmid	13
Luis Macedo	145	Ulrike Schmuntzsch	255
Tamas Madl	283	Michael Schoelles	66
Pierre Mayenobe	59	Tanja Schultz	303
Katja Mehlhorn	163	Dominik Seuss	13
Joachim Meyer	324	Michael Siebers	13
Johannes Meyer	303	Ingo Siegert	123
Esther Mondragon	234	Ian Simpson	315

Author Index

Paul Smart	165	Iris van Rooij	111, 199
Hardy Smieszek	125	Marieke van Vugt	218
Rebekah Smith	137	Ivan Vankov	7
Tomasz Smolen	181	Wolf Vanpaemel	257
Gert Storms	167, 251, 257	Steven Verheyen	251
Loes Stukken	257	Wouter Voorspoels	257
Niels Taatgen	175, 211, 241	Steffen Walter	123
Thora Tenbrink	230	Todd Wareham	111, 142, 311
Rick Thomas	31	Anne Wegerich	127
Makoto Toyota	291	Andreas Wendemuth	123
Greg Trafton	151	Robert West	72
Harald Traue	123	Sharon Wood	49
Akira Utsumi	243	Ulas Yilmaz	127
Fons Van Assche	167	Richard Young	55
Leendert van Maanen	273	Erica Yu	169
Bram Van Rensbergen	167	Meng Yu	94
Jacolien van Rij	101	Ulaş Yılmaz	255
Hedderik van Rijn	101, 241, 273	Changkun Zhao	117

Reviewers

Eduardo Alonso	Markus Knauff	Rainer Reisenzein
Erik Altmann	Boicho Kokinov	Frank Ritter
Thierry Bellet	Joseph Krems	Nele Rußwinkel
Jelmer Borst	Bernd Kroeger	Dario Salvucci
Duncan Brumby	Johan Kwisthout	Ute Schmid
Mike Byrne	Peter Lane	Mike Schoelles
Rick Cooper	Christian Lebiere	Lael Schooler
Gary Cottrell	Rick Lewis	Barry Silverman
Fabio Del Missier	Luís Macedo	Robert St. Amant
Uwe Drewitz	Michael Matessa	Terrence Stewart
Wai-Tat Fu	Ralf Mayrhofer	Andrea Stocco
Danilo Fum	Alain Mille	Niels Taatgen
Kevin Gluck	Claus Moebus	Julia Taylor
Fernand Gobet	Shane Mueller	Greg Trafton
Cleotilde Gonzalez	Christopher Myers	Leon Urbas
Glenn Gunzelmann	Josef Nerb	Leendert van Maanen
Tim Halverson	Hansjoerg Neth	Marieke van Vugt
Andrew Howes	David Noelle	Robert West
Christian Janssen	David Peebles	Sharon Wood
Gary Jones	Thad Polk	Richard Young
Mark Keane	Marco Ragni	