Leveraging Novel Information Sources for Protein Structure Prediction

vorgelegt von
Dipl.-Biol.
Michael Bohlke-Schneider, geboren Schneider
geb. in Dortmund

von der Fakultät IV — Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften
- Dr. rer. nat. -

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr. Klaus Obermayer
1. Gutachter:	Prof. Dr. Oliver Brock
2. Gutachter:	Prof. Dr. Juri Rappsilber
3. Gutachter:	Prof. Dr. Jens Meiler

Tag der wissenschaftlichen Aussprache: 22.12.2015

Berlin 2016

I would like to dedicate this thesis to my loving wife Nina and my mother Barbara.

Acknowledgements

Becoming a scientist was clearly the greatest challenge that I had to master in my life. The past five years changed my views and my life in a way that I never thought to be possible. Many people helped me during this journey. I am blessed with having many good friends and I cannot possibly list all the people that I am grateful for.

First, I want to thank my adviser, Oliver Brock. Oliver is the sharpest mind I have ever met and his ability to think incredibly clearly has been very inspiring. In addition, Oliver has redefined my standard for scientific rigor and what it means to master all skills that a scientist should have. But his most important lesson was to never be satisfied, to constantly challenge the *status quo*, and to change the world for the better. This lesson will surely shape my further life, no matter where it will take me.

I would also like to thank the members of my committee, Juri Rappsilber and Jens Meiler. Juri's dedication in our cooperation project contributed a great deal to my scientific success. I also enjoyed the numerous insightful and fun conversations with Juri. I appreciate all the time that he took to advice me during my graduate studies.

I would like to thank Jens Meiler for becoming a member of my committee. I appreciate his time reading my thesis and making it possible to arrange the defense at very short notice.

Lila Gierasch and Anne Gershenson guided me through my first years in graduate school. I would like to thank them for their advice.

The next important block of people is clearly the amazing crowd that I spent so much time with at the Robotics and Biology Laboratory. This includes (in no particular order): Ines Putz, Raphael Deimel, Arne Sieverling, Roberto Martin Martin, Vincent Wall, Tim Werner, Mahmoud Mabrouk, Rico Jonschkowski, Sebastian Höfer, Clemens Eppner, Jessica Abele, Manuel Baum, and Fabian Heinemann. If anything, you guys were the decisive factor for my success as a graduate student. All the amazing people at RBO created a fun and vibrant community with an honest and inspiring atmosphere that I have never experienced before. Thank you for all the critique, help, and warm words along my way.

I would also like to thank Alexander Margraf and Wolf Schaarschmidt for all the technical support and chats. Thank you Janika Urig for holding the lab together, being the good spirit, and for being a great listener.

I also thank Adam Belsom and Lutz Fischer for their amazing work and productive cooperation.

Furthermore, I thank "the other people" on the fifth floor at Marchstrasse that contributed to the great atmosphere: Robert Lieck, Johannes Kulick and Marianne Maertens.

I would also like to thank the people that I spend only little time with, but nonetheless appreciated: Dov Katz, Ingo Kossyk, José Antonio Álvarez Ruiz, Nasir Mahmood, Florian Kamm and Georgios Fagogenis.

Special thanks also goes to the members of the Budisa lab that I spend a lot of time with at parties and the Schleusenkrug. In particular, I would like to thank Maxi Marock and Jessica Nickling who are of such great help by caring for my dog Lilly when I cannot do it.

There have been many people that made my time in Berlin truly enjoyable. The following became close friends and I hope that we will never get out of touch because every one of them is a truly remarkable person: Bastian Henkel, Melanie Henkel, Kim Dohlich, Christian Busse, Claudia Luber, and Felix Luber. Thank you for joining me in good times and advising me at times of distress.

Of course, I have to thank all my important friends at home and those that are scattered throughout the world: Benjamin Tuma, Lukas Börger, Bastian Haumann, Stefan Pennartz, Timo Schulte, Paul Ratka, Pit Wingender, Philip Scheit and, Christian Saßenscheidt.

I also like to thank all the members of my karate club Shirokuma Berlin and the members of the Berlin board gaming community for many hours of fun. In particular: Rico Leifheit, Boris Mahn, and Heiko Möller.

An incredible load of thanks goes to my family. My mother Barbara Schneider always supported me in all my years of study and is the source of the wisest advice I have ever received. I also thank Heino Goldschmidt for bringing a new spark of life to my family. I thank my older sister Andrea Bielski for being incredibly supportive and my younger brothers Daniel Schneider and Markus Schneider. I always look forward to spent time with you. I would also like to thank my mother-in-law, Maria Bohlke-Wohlers, for spreading optimism wherever she goes.

The next person is not really a person but my dog Lilly. I thank her for making me happy whenever she is around.

My last acknowledgement is dedicated to my loving wife Nina Bohlke. Nina approaches anything in her life with incredible charm, wit, and warm-heartedness. Her determination and discipline are second to none. Nina naturally has the calm and wise mind that I hope to develop at some point in my life. I'm incredibly blessed for being able to call her my wife.

Prepublication and Statement of Contribution

This thesis has been in part published in the following publications (listed by date in chronological order, starting with the newest publication):

A Mabrouk, M., Werner, T., Schneider, M. Putz, I., and Brock, O. (2015) Analysis of Free Modeling Predictions by RBO Aleph in CASP11. Proteins, in press.

B Belsom, A.*, Schneider, M.*, Fischer, L., Brock, O., and Rappsilber, J. (2015). Serum Albumin Domain Structures in Human Blood Serum by Mass Spectrometry and Computational Biology. Mol. Cell. Proteomics, in print: mcp.M115.048504.

C Mabrouk, M.*, Putz, I.*, Werner, T., Schneider, M. Neeb, M., Bartels P. and Brock, O. (2015). RBO Aleph: leveraging novel information sources for protein structure prediction. Nucleic Acids Res., 43(W1):W343–W348.

D Schneider, M. and Brock, O. (2014). Combining Physicochemical and Evolutionary Information for Protein Contact Prediction. PLoS ONE, 9(10):e108438.

* contributed equally

Chapters 1, 2, and 4 are original to this thesis.

Chapter 3 contains a discussion of the related work. Parts of this chapter have been published in the related work discussion of paper [A, B, C, D].

Chapter 5 presents a contact prediction algorithm called EPC-map and is based on the original publication [D].

Own contributions to [D]: I (MS) was the sole first author of this paper. I conceived and designed the experiments, conceived and designed the algorithm, implemented the algorithm,

performed experiments, developed analysis tools, analyzed the data, and contributed to paper writing.

Contributions of co-authors to [D]: OB scientifically advised this work, conceived and designed the experiments. OB contributed to paper writing.

Chapter 6 presents the analysis of the structure prediction server RBO Aleph in the 11^{th} community-wide Critical Assessment of Protein Structure Prediction experiment (CASP11). RBO Aleph server was originally described in [C]. The analysis of the CASP11 results of this server was originally published in [A].

Own contributions to [C]: My particular contribution was the design and implementation of EPC-map and contact-guided model-based search (main developer from 2010-2015) into the *ab initio* pipeline of RBO Aleph. I also conceived and designed the server. I designed and implemented significant parts of the server, especially on the backend. I maintained the server during CASP11 and contributed to paper writing.

Contributions of co-authors to [C]: MM, IP, TW, and OB conceived and designed the server. MM, IP, TW designed and implemented significant parts of the server. MM, IP, TW, PB, and MN designed and implemented the frontend of the web server. MM, IP, and TW maintained the server during CASP11. MM, IP, TW, and OB contributed to paper writing.

Own contributions to [A]: I conceived and designed experiments. I performed experiments and developed analysis tools. I analyzed the data, with a focus on the residue-residue contact prediction results of RBO Aleph in CASP11 and the effect of component combinations on the pipeline output. The contact prediction results of RBO Aleph are based on the algorithm developed in chapter 5. I was the main developer of the conformational space search algorithm of RBO Aleph, contact-guided model-based search, from 2010-2015. I contributed to paper writing.

Contributions of co-authors to [A]: MM, TW, IP, and OB conceived and designed experiments. MM, TW, and IP performed experiments. MM, TW, and IP conceived and implemented analysis tools. MM, TW, and IP analyzed data. MM, TW, IP, and OB contributed to paper writing. The following figures and tables were prepared in part or in modified form by co-authors of the original paper [A]: MM: Figure 6.6, 6.9. TW: Figure 6.7, 6.8, 6.11, 6.12, Table 6.4. IP: Table 6.1.

Chapter 7 presents a novel hybrid structure determination method based on high-density cross-linking/mass spectrometry and conformational space search. This chapter is based on paper [B].

Own contributions to [B]: I conceived and designed the experiments. In particular, I conceived and designed the cross-link-guided conformational space search protocol. I implemented the cross-link-guided conformational space search protocol and performed all protein modeling experiments. I contributed analysis tools and analyzed the data, with a focus on the analysis of the structure determination results. I contributed to paper writing.

Contributions of co-authors to [B]: AB, LF, JR, OB conceived and designed experiments. AB performed experiments, in particular experimental cross-linking/mass spectrometry. LF contributed data analysis tools, in particular analysis of cross-link data. AB and LF analyzed data. AB, LF, JR, and OB contributed to paper writing. The following figures and tables were prepared in part or in modified form by co-authors of the original paper [B]: AB: Figure 7.1, 7.2, 7.3, 7.4, 7.5, 7.8, 7.9. LF: Figure 7.6, 7.7.

Chapter 8 is original to this thesis.

Own contributions to chapter 8: I conceived and designed the experiments, conceived and designed the algorithm, implemented the algorithm, performed experiments, developed analysis tools, and analyzed the data.

Contributions of co-authors: The algorithm presented in chapter 8 was conceived by me (MS), Juri Rappsilber, and Oliver Brock at a research meeting at the University of Edinburgh. OB and JR scientifically advised this work.

Chapter 9 concludes this thesis. Some of the conclusions have also been drawn in the original papers [A, B, C, D].

Appendix A introduces evaluation criteria, some of which were originally introduced in [D].

Appendix B contains a detailed feature description of EPC-map that was originally part of the supporting information of [D].

Appendix C contains PDB IDs of the training set and test set used in [D].

Abbreviations

AA	Amino acid
Acc	Accuracy
AUROC	Area Under ROC-curve
BLAST	Basic Local Alignment Tool
BS3	Bis(sulfosuccinimidyl)suberate
CASP	Critical Assessment of Protein Structure Prediction
CLMS	Cross-linking/mass spectrometry
CRF	Corticotropin-releasing factor
Cryo-EM	Cryo-electron microscopy
DNA	Deoxyribonucleic acid
DSSP	Define Secondary Structure of Proteins
EPC-map	Using Evolutionary and Physicochemical information to predict Contact maps
EPR	Electron paramagnetic resonance
FDR	False discovery rate
FPR	False positive rate
FM	(Template) Free modeling
fMRI	Functional magnetic resonance imaging
GDT_TS	Global distance test/total score
GPCR	G-protein coupled receptor
HMM	Hidden markov model
HSA	Human serum albumin
LC-MS/MS	Liquid chromatography-tandem mass spectrometry
MBS	Model-based search
MC	Monte Carlo
MeCP2	Methyl-CpG-binding protein
MPI	Message passing interface
MSA	Multiple-sequence alignment
NHS	N-Hydroxysuccinimide
NMR	Nuclear magnetic resonance
PDB	Protein Data Bank

Photo AA	Photo-amino acid
PSM	Peptide spectrum match
RMSD	Root-mean-square deviation
ROC	Receiver operator characteristic
SIFT	Scale Invariant Feature Transform
sSASD	Shortest solvent accessible surface distance
sulfo-SDA	Sulfosuccinimidyl 4,4'azipentanoate
SVM	Support vector machine
TBM	Template-based modeling
TM-score	Template-modeling score
TPR	True positive rate
UV	Ultraviolet

Abstract

Three-dimensional protein structures are an invaluable stepping stone towards the understanding of cellular processes. Computational protein structure prediction holds the promise of providing these structural models at low cost and effort. However, the major bottleneck towards effective protein structure prediction is the high dimensionality and vast size of the protein conformational space. These properties of the conformational space make it extremely difficult to locate the native structure through search. Information alleviates this issue by guiding search towards the native protein structure. Thus, information is invaluable in conformational space search.

Not surprisingly, state-of-the-art structure prediction methods heavily rely on information. Obviously, unlocking novel sources of information should further increase our ability to accurately predict protein structure.

This thesis leverages three *novel* sources of information to advance protein structure prediction. First, we leverage **physicochemical information** that is encoded in energy functions and predicted structure models. Native contact networks form characteristic patterns to be energetically favorable. This thesis develops a network-based representation to capture these patterns and uses this representation to predict residue-residue contacts.

The second source of information is experimental data from **high-density cross-linking/ mass spectrometry (CLMS)** experiments. We integrate this information in an experimental/computational hybrid method for protein structure determination.

The third information source is **corroborating information**. Corroborating information judges the likelihood of the co-occurence of structural constraints. Nearly all methods provide these constraints in isolation, thereby neglecting any corroborating evidence between them. We develop a network-based analysis method to refine structure constraints with corroborating information.

We demonstrate the value of these information sources in extensive *ab initio* structure prediction experiments with a customized conformational space search algorithm and a novel structure prediction pipeline. This pipeline reached state-of-the-art contact and *ab initio* structure prediction performance in the 11^{th} community-wide Critical Assessment of Protein Structure Prediction experiment (CASP11).

Using our CLMS-based hybrid method, we reconstruct the domain structures of human serum albumin in solution and in its native environment, human blood serum. This represents a disruptive first step towards a mass spectrometry-driven, *ab initio* structure determination method that is able to probe protein structure where it really matters: In their natural environment, which is their very place of action.

Zusammenfassung

Die Kenntnis von dreidimensionalen Proteinstrukturen ist für das Verständnis von zellulären Prozessen unverzichtbar. Computergestützte Verfahren zur Proteinstrukturvorhersage haben das Potenzial diese strukturellen Modelle mit wenig Aufwand und niedrigen Kosten zu generieren. Allerdings ist die hohe Dimensionalität und schiere Größe des Konformationsraumes ein großes Hindernis auf dem Weg zur effektiven Strukturvorhersage. Diese Eigenschaften des Suchraumes machen es extrem schwierig die native Proteinstruktur mittels Suchalgorithmen zu finden. Information leitet die Suche nach der nativen Struktur. Daher ist Information für die Suche im Konformationsraum unverzichtbar.

Viele Proteinstrukturvorhersagemethoden nutzen ein hohes Maß an Information. Offensichtlich sollte das Erschließen neuer Informationsquellen unsere Fähigkeit zur genauen Strukturvorhersage massiv erweitern.

Diese Dissertation demonstriert den Einsatz drei neuartiger Informationsquellen in der Strukturvorhersage.

Die erste Informationsquelle ist physikalisch-chemische Information, enthalten in Energiefunktionen und vorhergesagten Strukturmodellen. Native Kontakte bilden charakteristische Netzwerke aus, um energetisch günstig zu sein. Diese Dissertation entwickelt eine Netzwerk-basierende Repräsentation dieser charakteristischen Netzwerke um Proteinkontakte vorherzusagen.

Cross-link/Massenspektrometrie (CLMS) Daten mit extrem hoher Dichte sind die zweite Informationsquelle. Wir integrieren diese Information in einer experimentellen/ computergestützten Hybridmethode für die Strukturbestimmung.

Die dritte Informationsquelle sind sich unterstützende Informationen. Diese beurteilen die Wahrscheinlichkeit vom simultanen Auftreten mehrerer struktureller Zwangsbedingungen. Nahezu alle Methoden sagen diese Zwangsbedingungen isoliert vorher und ignorieren daher unterstützende Informationen. Wir entwickeln eine Netzwerkanalysemethode um mit dieser Information Zwangsbedingungen zu verfeinern.

Wir demonstrieren den Nutzen dieser Informationsquellen in umfangreichen *ab initio* Strukturvorhersageexperimenten mit einem modifizierten Suchalgorithmus und eines neuartigen Strukturvorhersagesystems. Mit diesem System waren genaue Kontaktvorhersagen

und *ab initio* Strukturvorhersagen in dem elften „Critical Assessment of Protein Structure Prediction“ Experiment möglich. Mit unserer CLMS-basierenden Hybridmethode konnten wir die Struktur der Domänen von Humanalbumin rekonstruieren. Dies war für isoliertes Humanalbumin und für Humanalbumin in Blutserum möglich, welches die natürliche Umgebung dieses Proteins darstellt. Dies ist ein wichtiger erster Stritt in Richtung einer neuen CLMS-basierenden Strukturbestimmungsmethode. Diese ist in der Lage strukturelle Informationen da zu sammeln wo es wirklich darauf ankommt: In der natürlichen Umgebung von Proteinen, in welchen sie ihre Funktion ausüben.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis addresses the problem of protein structure prediction, which is the prediction of the three-dimensional structure from the sequence of a protein. Finding the native structure of a protein requires search in a rugged energy landscape that contains many local minima. This problem is exacerbated by the high-dimensionality of the search space because the size of the space grows exponentially with each added dimension.

If we assume two degrees of freedom and ten discrete conformational spaces per amino acid, a protein with 150 amino acids has 10^{300} possible states. Even an earth-sized computer that operates at the Bremermann limit[1] [25] (10^{75} operations per second) would need approximately 10^{217} years to exhaustively enumerate all conformational states of this protein. Thus, exhaustive search cannot address search problems of this size.

The problem of search in high-dimensional spaces is not unique to protein structure prediction and scientists from different fields acknowledged this problem in their respective domain. Richard Bellman originally conceived the notion of the "curse of dimensionality" for high-dimensional search problems [9][2]. In the domain of protein folding, the problem of high dimensionality is known as the "Levinthal paradox": There are too many possibilities for a medium-sized protein to find the native structure by random search [117]. Yet, real proteins fold in a few seconds. Thus, there must be another way for the protein to find its native structure.

Since exhaustive search in high-dimensional spaces is impossible, and in the case of protein folding also implausible, we need information to steer search towards the solution. In fact, information is often the decisive factor for effective search.

[1] Bremermann estimated the physical limitation on the progress of computation: "No data processing system whether artificial or living can process more than $2 * 10^{47}$ bits per second per gram of its mass" [25].

[2] Bellman used this argument to motivate his favored approach of dynamic programming. However, dynamic programming imposes additional constraints on the search space, such as optimal substructure. Many real world problems do not have this property and therefore cannot be solved by dynamic programming.

We now give three examples of search problems from various domains that address the problem of high-dimensional space search by leveraging varying degrees of information.

1. **Computer vision:** The Scale Invariant Feature Transform (SIFT) approach in computer vision extracts image features that are invariant to affine distortion, change in viewpoint, and illumination [124]. Thus, this approach implicitly leverages information about the problem domain: Differences in images that originate from physical phenomena, such as illumination, should not influence the outcome of image matching and object recognition.

2. **Machine learning:** Another example comes from the domain of machine learning. The careful construction of features is widely acclaimed to be the key of powerful machine learning algorithms [48]. Understanding about the problem domain guides feature engineering. Thus, machine learning researchers inject information into the algorithm through the feature engineering process by distilling problem-specific knowledge into features.

3. **Protein structure prediction:** Interestingly, Zwanzig et al. [243] showed in a mathematical model of protein folding that the mean-passage time of a folding event is reduced from 10^{27} years to one second if a moderate bias to locally incorrect conformations is introduced. This suggests that the introduction of bias, which can also be viewed as information, resolves the Levinthal paradox. In fact, the most effective protein structure prediction algorithms introduce a "bias" to search: They leverage information in the form of local structural fragments and template structures [190, 195].

These three examples from different problem domains demonstrate that information makes search in high-dimensional spaces feasible. Thus, we argue that an effective algorithm for high-dimensional space search needs to successfully leverage information and convert it into the solution.

In structure prediction, the most effective algorithms to date, fragment assembly and homology modeling, clearly follow the principle of leveraging information. Fragment assembly algorithms leverage information about local protein structure and use Monte Carlo sampling to convert this information into the native structure [190]. Homology modeling algorithms leverage information about the structure of related proteins by identifying them by sequence matching and converting this information into the solution with distance geometry [4, 195].

These two algorithms lead to the biggest boosts in structure prediction accuracy [146, 147]. Virtually all current structure prediction algorithms are refined instances of these approaches and use fragments and/or templates. Because no effective type of information

has been discovered since fragments and templates, the field of protein structure prediction progresses very slowly since the introduction of these concepts [108][3].

In this thesis, we argue that protein structure prediction is advanced by leveraging *novel* information sources. Since progress in structure prediction is driven by information, using novel information sources should increase prediction accuracy. In particular, we focus on unlocking information sources that are orthogonal to existing information.

1.1 Contributions

The **first contribution** in this thesis is a graph-based machine learning algorithm to predict residue-residue contacts from search space samples, i.e. *ab initio* protein structures. These structures are low-energy states in an energy landscape and thus contain physicochemical information that is encoded by the energy function and the protein structure. We extract this information by a graph-based encoding that captures the physicochemical information inherent in *ab initio* protein structures. We then employ machine learning to predict contacts. We combine physicochemical contacts with evolutionary contacts to further improve prediction accuracy. We verified our contact prediction approach in the 11^{th} Critical Assessment of Protein Structure Prediction (CASP). Our method ranked second for medium+long range contacts and fifth for long-range contacts. In addition, our contact prediction method significantly contributed to the free-modeling performance of our tertiary structure prediction server, RBO Aleph. Out of 44 automated methods, RBO Aleph ranked first by average Z-score > 0 and third by sum Z-score > -2 (ranking of the first model by the assessors' formula).

The **second contribution** is a structure determination method that leverages novel, high-density cross-linking/mass spectrometry (CLMS) data, stemming from the heterobifunctional cross-linking reagent, sulfosuccinimidyl 4,4'azipentanoate (sulfo-SDA). We develop an iterative conformational space search strategy that exploits information from CLMS data to determine protein structure. We reconstruct the domain structures of human serum albumin (HSA) to verify this approach. Importantly, cross-linking can be performed in complex biological environments because the chemical reaction is not disturbed by the presence of other molecules and mass spectrometry is able to read out the information from complex

[3]The recent development of new algorithms to evolutionary contact prediction are widely believed to have the potential to become the next important information source in structure prediction [103, 132, 142, 152]. However, evolutionary contact prediction needs deep sequence alignments. Unfortunately, templates are often available for proteins with deep alignments, which makes evolutionary contacts redundant for these proteins because their structure can already be predicted using homology modeling [200]. Although recent studies support the value of evolutionary contacts in structure prediction [104, 138], their applicability to the general structure prediction scenario remains to be proven [42].

mixtures[4]. This allows us to reconstruct the structure of HSA domains from in-serum samples, the natural environment of this protein. This is an important stepping stone towards probing protein structure where it really matters: In their native environment.

The **third contribution** is the development of algorithmic strategies to refine contact and CLMS data. Contact and CLMS data represent residue-residue constraints in the form of Euclidean upper distance bounds. The methods in this thesis derive these constraints individually. Thus, knowledge about the likelihood of particular constraint patterns is typically ignored. Given a set of constraints, some constraints reinforce each other by forming self-consistent sets. We leverage this corroborating evidence between constraints using network analysis. We demonstrate that this approach increases the accuracy of residue-residue contact and CLMS data.

1.2 Thesis Structure

This thesis is organized as follows: Chapter 2 introduces the application domain of protein structure prediction. We focus our introduction on the information sources that are employed to solve the structure prediction problem. Chapter 3 reviews the related work. Again, we focus on algorithms that leverage information for protein structure prediction. Chapter 4 introduces the technical background in network analysis, machine learning, and cross-linking/mass spectrometry. We recommend this chapter if the reader is unfamiliar with these topics. In chapter 5, we introduce our approach to contact prediction from protein decoys with physicochemical information (EPC-map). Chapter 6 presents the verification of this approach in the CASP11 experiment. In chapter 7, we introduce a novel protein structure determination method that combines high-density cross-linking/mass spectrometry data with conformational space search. With this method, we reconstruct the domain structures of human serum albumin from protein samples in complex biological matrices. Chapter 8 describes our approach of refining structural constraints with corroborating evidence. Chapter 9 concludes this thesis and outlines future work. Appendix A summarizes evaluation criteria used in contact and structure prediction. Appendix B lists the features used in EPC-map. Appendix C lists the training set and test set of EPC-map that were constructed for this thesis.

[4]Note that cross-links can indeed react with molecules other than the target molecule, but this signal can eventually be separated by sample preparation and mass spectrometry.

Chapter 2

Protein Structure Prediction

This chapter provides an overview of protein structure prediction. We focus on the driving role of information sources that advance protein structure prediction. Section 2.1 discusses the significance of protein structure prediction for structural biology. Section 2.2 reviews the role of information in the history of the Critical Assessment of Protein Structure Prediction (CASP) experiment. The CASP history shows that protein structure prediction is driven by the discovery and exploitation of information. Thus, we discuss different types of information in structure prediction in section 2.3. In section 2.4, we summarize how these types of information are used in structure prediction. We conclude this chapter with a discussion of hybrid methods, which leverage additional information from experimental data 2.5.

2.1 The Significance of Protein Structure Prediction

The three-dimensional structure of biological macromolecules is invaluable to understand cellular processes. The DNA double helix structure of Watson and Crick spurred studies of the molecular mechanism of DNA replication [8]. Recently, researchers found new ways to determine the structure of membrane proteins. This lead to unprecedented insight into the molecular mechanisms of G-protein coupled receptors, a protein family that detects numerous extracellular signals. Insight into GPCR function is of high biomedical interest, since GPCRs are key components of signal cascades and therefore primary drug targets for treating cardiovascular diseases and neurological disorders [206]. The ribosome is the key machinery in biological protein synthesis. Thus, many antibiotic drugs target the ribosome. Structural studies on the ribosome in complex with antibiotic drugs revealed the mechanisms of antibiotic function and the mechanisms of bacterial resistance [219]. Insights from ribosome structures drive research efforts to combat emerging antibiotic resistance, an increasingly severe public health threat that the World Health Organization predicts to cause

"increased morbidity, prolonged illness, a greater risk of complications, and higher mortality rates." [222].[1]

The importance of protein structure determination is contrasted by the high cost and experimental effort that is required to solve a single structure by X-ray crystallography or nuclear resonance spectroscopy (NMR). The pressing need to scale up protein structure determination lead to the establishment of the Protein Structure Initiative project (PSI). PSI started in 2000 with the goal of high-throughput determination of novel protein folds. The PSI project solved more than 6,500 structures and contributed to the determination of challenging membrane protein targets. However, the project will be discontinued in 2015 [49]. Ultimately, and despite the accomplishments of the PSI consortium, the "big science" approach to protein structure determination did not create enough impact to justify the 70 million dollar per year cost of the project.

In addition, sequencing technology develops much faster than structure determination technology. This leads to an increasing gap of known protein structures and sequences. In 2012, the UniProt database [202] contained 20 million sequences and the protein structure databank 86,364 structures [12]. The number of sequences in the UniProt database has more than doubled from 2012 to 2015 (more than 50 million entries). In contrast, the protein structure databank only grew from ~86 thousand structures to about 100 thousand structures in the same time.

We view these facts —the importance of protein structure, the limited success in scaling up protein structure determination, and the ever increasing sequence-structure gap— as an indication that novel technologies are required to provide protein structures.

Computational approaches to protein structure prediction hold the promise of providing structural models at low cost and effort. Protein structure prediction is theoretically unlimitedly scalable and can compute structural models for any protein system. In addition, protein structure prediction algorithms can be complemented with experimental data to produce structural models that are underpinned by empirical information [216].

2.2 Information Drives Progress in Structure Prediction

The Critical Assessment for Protein Structure Prediction experiment (CASP) is an biannual experiment to test and analyze the state-of-the-art in protein structure prediction. In this experiment, the organization committee releases protein sequences of protein targets with

[1] Structural studies of these protein systems also resulted in several nobel prizes: Medicine 1962, Watson, Crick and Maurice; Chemistry 2012, Lefkowitz and Kobilka; Chemistry 2009, Ramakrishnan, Steitz and Yonath.

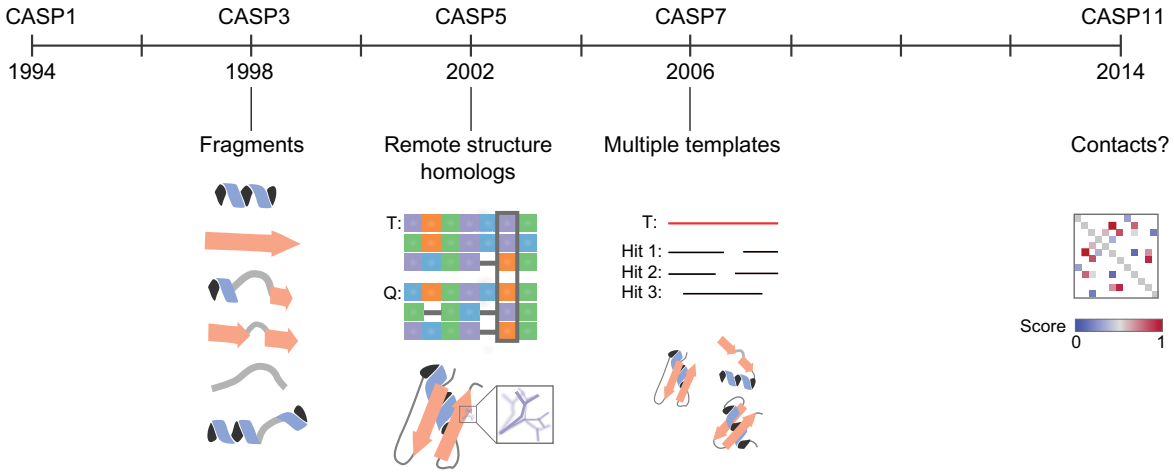

Fig. 2.1 Impact of information on the CASP experiment. The most successful types of information in the history of CASP were short structural fragments, remote structural templates from sequence profile-profile alignments, and the combination of multiple templates. In CASP11, residue-residue contacts emerged as a new type of information that impacted the experiment.

unknown structure. CASP participants predict the structures within a given time frame and submit the models to the CASP prediction center. Independent assessors evaluate the submitted predictions. This experiment is regarded as the gold standard in protein structure prediction and one of the most remarkable accomplishments in computational biology [175].

In CASP history, the unlocking and utilization of novel information sources drives progress in structure prediction (Figure 2.1). Hallmarks in CASP history are fragments in CASP3 [147], recognition of remote structural homologs in CASP5 [146], and the combination of information from multiple templates [144]. Since CASP7, the most successful predictors use sophisticated combinations of template-based modeling and *ab initio* methods [172, 196, 236]. However, significant progress has not been noted since CASP7. We see this as an indication that the information sources that drove progress is CASP exhausted their potential and momentum could be gained by leveraging other sources of information.

Current CASP experiments suggest that residue–residue contact prediction methods might have lasting impact in CASP. Although the idea of contact prediction as an intermediate step to structure prediction is not novel (one of the first algorithms was introduced in 1994 [69]), these methods got increased interest since the introduction of evolutionary methods. This approach predicts contacts from multiple-sequence alignments using global statistical methods and provide much more accurate contact maps than earlier methods [88, 131]. Pure evolutionary contact prediction needs large multiple-sequence alignments [132].

Thus, their modest impact in CASP is presumably due to the low number of sequences that are available for CASP targets [145].

Recent results suggest that current contact prediction methods have the potential to impact CASP. Kosciolek and Jones [105] report successful use of contacts in 3D structure prediction in CASP11[2]. The results presented in this dissertation also suggest that contacts increases *ab initio* structure prediction performance, albeit by using a different information source (physicochemistry).

Since the exploitation of information is a driving factor in protein structure prediction, prediction should improve if other information sources are successfully leveraged.

2.3 Information Defines Method Categories in Structure Prediction

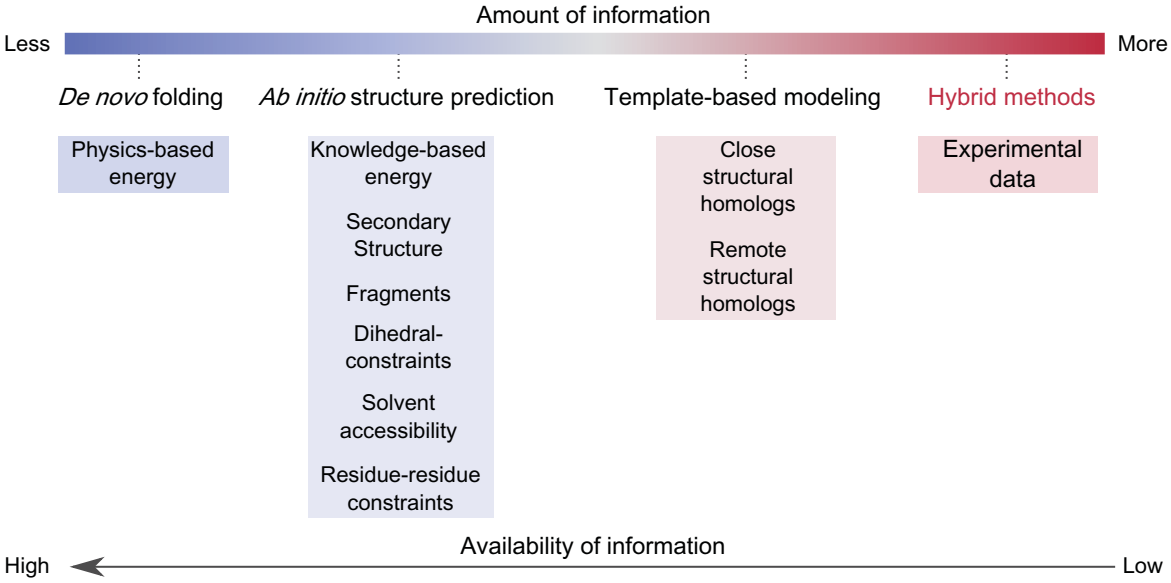

Fig. 2.2 Categorization of protein structure prediction methods. Researchers categorize structure prediction methods by the type and amount of information that they use in structure modeling. These categories are (listed from least information used to most information used): *de novo* folding, *ab initio* structure prediction, template-based methods, and hybrid methods. The accuracy of the methods increases with the amount of available information. Thus, practitioners select their employed method by the available information.

[2]Note that the algorithm by Kosciolek and Jones [105] uses sequence-based machine learning and evolutionary contacts. Therefore it is strictly speaking not a pure evolutionary contact prediction method and leverages additional information through machine learning.

In the previous section, we showed that information is the driving factor in the CASP experiment. Not surprisingly, researchers also categorize protein structure prediction methods by the amount and availability of information (Figure 2.2). Practitioners typically prefer the method that uses most information to predict protein structure, since this method usually results in the most accurate model. Hence, the decision of which method is used for modeling is driven by the availability of information.

***De novo* folding** methods only use physics-based energy terms. These methods start with the unfolded protein chain and sample the conformational space by molecular dynamics using Newton's laws of motion [153, 183].

***Ab initio* structure prediction** methods use additional information, such as knowledge-based energies, secondary structure, fragments, dihedral constraints, and residue-residue constraints. Within the algorithmic workflow, these information types are predicted prior to 3D modeling and are used as input to the structure prediction algorithm. These methods usually search the conformational space with Monte Carlo sampling [87, 170, 227].

Template-based modeling methods use structures from close or remote homologs [4, 78, 158, 171, 195, 225, 230]. If the quality of the template match is high enough, the backbone conformation of the resulting model corresponds to the template conformation. Otherwise, the structure of the model is computed by distance geometry [4]. However, these methods also model the side-chains of the target sequence [107]. In difficult template-based modeling cases the template match quality is often low or the target sequence is only partially covered by the template. In such cases, the structure model can be completed with *ab initio* structure prediction methods [172, 196, 236].

Hybrid methods This category of methods uses additional information from experimental data that is not available by prediction. Hybrid methods use *de novo* folding, *ab initio* structure prediction, or template-based modeling to predict protein structure. These methods use experimental data as constraints in search, to disambiguate structural homologs, and for validation of the model structure (see section 2.5).

Note that information content from *de novo* folding to hybrid methods does not only increase by using additional information, but also by using information from the "lower level". For example, *ab initio* structure prediction methods also use physics-based energy terms in hybrid energy functions [170]. Fragment picking algorithms select fragments with

the help of predicted secondary structure [43, 73, 94]. Furthermore, predicted secondary structure and solvent accessibility is also used to retrieve templates [78, 158, 230]. Finally, researchers complete template-based structural models using *ab initio* structure prediction algorithms [172, 196, 236].

2.4 Information Types in Protein Structure Prediction

We discussed that information drives progress in structure prediction and that their availability defines the categories of structure prediction methods. In this section, we discuss the different categories of information in more detail. We discuss information types that are derived from purely computational approaches. In section 2.5, we give an overview of information types from experimental data.

Please note that the distinction between pure computational approaches and methods that employ experimental data is not clear cut. Strictly speaking, physical experiments derived all available structures and sequences. Therefore, all methods that employ structure and sequence databases use experimental data in some form. For example, homology modeling uses experimental data from template structures tackle the structure prediction problem. To differentiate pure computational approaches from hybrid methods, we use the following definitions: 1) A pure computational approach to structure prediction uses computation or statistical analysis on sequence and structure databases without performing a physical measurement on the specific protein. 2) A hybrid method uses data from physical measurements on the specific protein in addition to computational structure prediction approaches.

We group computational information types by their amount of information into energies, local constraints, non-local constraints, and templates (Figure 2.3). Energies contain least information and require search to find the native structure. Templates are on the other side of the spectrum. A close homologous template contains enough information to model the native structure of the target protein. Local and non-local constraints contain explicit structure information (unlike energies) but still require search to find the native structure.

We now give a more detailed explanation of the information types in structure prediction.

2.4.1 Energies

Energies are objective functions that score protein conformations. Energy functions do not contain information that is specific to the conformation of the target protein. Instead, energy functions encode the physical potential energy of a molecular system by a set of functions

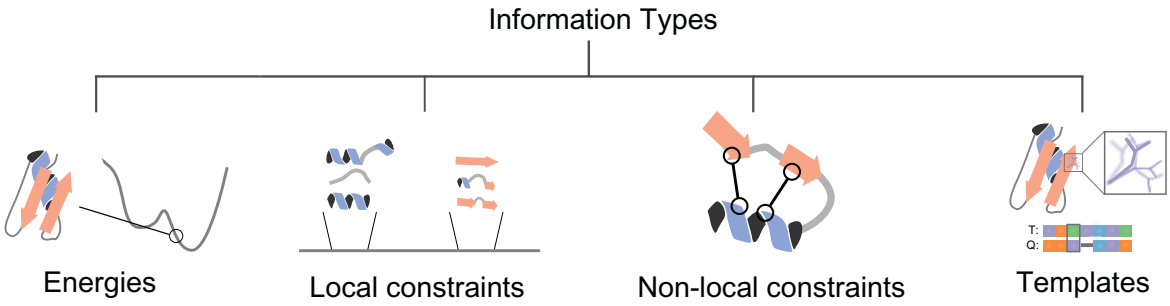

Fig. 2.3 Computational information types in protein structure prediction. Energies only contain implicit structure information and need to be complemented with conformational space search to predict protein structure. On the other end of the spectrum are templates, which can be sufficient to model the target structure. Local constraints and non-local constraints require search for structure lie between energies and templates on this spectrum of required conformational space search.

and parameters. Energy functions implicitly contain structure information because native structure features tend to be energetically favorable.

Energy functions are either physics-based or knowledge-based. Physics-based energy functions employ physical principles to score a conformation [151, 163]. Knowledge-based energy functions convert statistics in high-resolution structures into probabilities which are then used to score the likelihood of a conformation by Bayesian statistics [190, 193]. Energy functions can be defined for all-atom representations of the protein or for reduced representations where the side-chain atom is represented by a virtual "centroid" atom [187, 190]. Structure prediction algorithms perform the majority of conformational sampling in the reduced representation because energy evaluations on reduced representations are computationally more efficient.

Energy functions contain local, non-local, and global energy terms, such as reference energies of amino acids, electrostatics between residue-pairs, and reference values for the radius of gyration [151, 163, 170, 187, 190, 193].

Energy functions are only approximations of the real physical energy. They are therefore uncertain and conformational search sometimes finds non-native conformations that are lower in energy than native conformations [26, 203]. Energy functions are also ambiguous because multiple conformations might have approximately the same energy (degenerate energy states). Since energy function only score the current conformation and provide no guidance towards specific conformations, they need to be complemented with search to predict protein structure.

2.4.2 Local Constraints

Constraints reduce the conformational space by favoring (or penalizing) certain properties of a conformation. Unlike energies, constraints contain explicit (mostly geometric) information of the target protein. Researchers predict constraint information from the target sequence using sequence matching, statistical methods and/or machine learning [73, 86]. Local constraints contain information about the conformation of individual residues or about small, local backbone chunks. Local constraints include secondary structure, fragments, dihedral constraints, and solvent accessibility. Secondary structures are categorical constraints that capture the secondary structure content of a local part of the protein chain [90]. Fragments are short stretches of local structure that are excised from other proteins in the protein data bank [73]. Dihedral constraints define a range for the backbone dihedral angles around the protein chain. Secondary structures might also be considered as dihedral constraints, since each secondary structure category constraints the structure to the allowed values of the ramanchandran plot [165]. However, dihedral constraints can have a tighter range of valid dihedral values than secondary structure constraints.

Local constraints are typically not perfectly predicted. They are therefore uncertain and might contain prediction errors. Prediction errors in local constraints, especially errors in secondary structure prediction, can be a severe impediment to *ab initio* structure prediction [235]. In any case, local constraint information still require significant conformational space search to predict protein structure. However, the amount of required search is much lower compared to energies, because constraints guide search to specific regions of the conformational space.

2.4.3 Non-Local Constraints

Non-local constraints define a valid distance range for residue-residue pairs. Similar to local constraints, non-local constraint prediction methods employ statistical methods and machine learning [120, 131] (see section 3.1 for a detailed review of non-local constraint prediction).

Residue-residue constraints are exact distance values or contacts, which are residues that are below a certain distance threshold [143]. Researchers use various definitions for the atom type and the distance cutoff to define a contact. The most common definition for a residue contact is that two residues are in contact if the distance between their C_β atoms is below 8 Å [143].

Non-local constraints contain noise because not all predicted constraints are correct. Thus, an error tolerant representation of the non-local constraints is vital to use them effectively in structure prediction [104, 138]. They are additionally sparse because current methods only

predict a subset of all true contacts[3]. Non-local constraints represent spatial information about the protein structure that is highly valuable in structure prediction. Thus, combining non-local with constraints typically improves backbone accuracy [104, 138].

2.4.4 Templates

Structural homologs can be used as templates to model the protein structure. If a template is accurate enough (close homologs), the structure can be modeled by simply building the target backbone from the template restraints and eventually re-building the side-chain conformations [4, 107]. The sequences of the target protein and close homologs are typically very similar.

Remotely homologous proteins diverged earlier during evolution and therefore have dissimilar sequences (sequence similarity below 20% sequence identity) [195]. The information from remote homologs might be ambiguous, because the score from template retrieval programs is less reliable for remote homologs and therefore can lead to the selection of the wrong template [171]. In addition, the sequence of the homologous protein needs to be aligned to the target protein. Because of the low sequence identity, the sequence alignment of remote homologs can be challenging, which adds uncertainty to the modeling process [130].

In many cases, template information is sufficient to model protein structure with good backbone accuracy [239]. Nevertheless, ambiguity in template selection, alignment errors, and incomplete coverage of the target sequence might result in incomplete or erroneous information. In these cases, conformational space search complements template-based modeling [172, 196].

2.4.5 Other Information Types

Since the emphasis of this thesis is *ab initio* structure prediction, we focus on information types that contain information about the protein backbone. However, we want to briefly mention two other information types that are routinely used in structure modeling. Rotamer libraries capture the most probable side-chain conformations as observed in high-resolution crystal structures [182]. Rotamer libraries are an effective way to sample side-chain conformations and are often used as a last step in template-based protein modeling. Another type of information we do not discuss in detail are domain boundaries. Protein chains can be comprised of multiple domains. Domains are distinct units that fold and evolve on their own.

[3]To be precise, methods for contact prediction can provide a probability or score for all entries in the contact matrix. However, usually only a fraction of the predicted contacts is considered because the accuracy of the entire contact map is typically to low. Using only this fraction of top ranking contacts makes contact prediction sparse.

The problem of predicting the long protein chains is simplified by first splitting the chain into smaller domains [28, 51, 228].

2.5 Hybrid Methods

In this chapter, we argued for the importance of information in protein structure prediction and that protein structure prediction is driven by the successful exploitation of information. So far, all the information types we discussed have computational origin (see our definition of pure computational approaches to structure prediction on page 10). However, there is another important information type that originates from experiments on a physical sample of that specific protein.

These experimental methods infer structural constraints from physical phenomena. Researchers developed hybrid methods that integrate data from several experimental approaches, such as sparse backbone NMR [166], electron paramagnetic resonance [3, 79], small-angle X-ray scattering [61], low-resolution electron density maps [106, 121], Förster resonance energy transfer [18], and cross-linking/mass spectrometry [36, 80, 97, 162, 191, 231].

Information from these methods alone is usually not sufficient to compute the structural model, in contrast to established structure determination techniques, such as X-ray crystallography and NMR spectroscopy. Thus, information from these methods needs to be complemented with physical theories, computation, and with information from sequence and structure databases [216][4]. The approach of incorporating experimental data into the structure prediction process is called "integrative methods" or "hybrid methods"[5].

Hybrid methods have three important benefits. First, the information from experimental data pinpoints the conformational space region of the native structure. This reduces the search space and improves the accuracy and completeness of the resulting model. Second, models resulting from hybrid methods are verified by empirical data. Verification by empirical data increases the belief into the structural model, especially if information from multiple sources is combined to build the model. Third, a number of experimental methods, such as FRET, CLMS, and in-cell NMR [173] are able to probe the structure of a protein in its native environment. Importantly, this allows hybrid methods to elucidate structures of proteins that are elusive to X-ray crystallography and/or NMR methods, because they cannot be purified, do not crystallize [174], or need their natural environment for correct folding.

[4] Low-resolution electron density maps and sparse NMR constraints are also not sufficient to determine the protein structure alone and need to be complemented with computation [123].

[5] All approaches to structural biology are integrative methods to some degree [174]. X-ray crystallography involves fitting a structural model into the electron density. Structure determination with NMR uses molecular dynamics and energy minimization to compute the resulting model.

Hybrid methods determined the structure of important protein systems, such as the lid of the proteasomal complex and the type III secretion needle [123, 162]. The proteasome is an biologically important protein system, which degrades proteins that are no longer needed or damaged by proteolysis [67]. Politis et al. [162] probed the structure of the proteasomal lid with a combination of native mass spectrometry, cross-linking, and Monte Carlo optimization of the complex structure with MS derived constraints. Pathogenic, gram-negative bacteria secrete effector proteins into eucaryotic cells with the type III secretion protein complex. Thus, the structure-function relationship of this system is of outstanding importance for infection biology. Researchers have not yet succeed to determine the structure of this system by X-ray crystallography, because of its insolubility and resilience towards crystallization. Loquet et al. [123] modeled this protein system by combining the Rosetta Fold&Dock protocol with a cryo-EM map and a sparse set of 521 constraints from solid-state nuclear magnetic resonance (NMR) experiments.

Evidently, hybrid methods are able to investigate the structure of biologically important molecules. See Ward et al. [216] and Sali et al. [174] for comprehensive reviews on hybrid methods. Since this dissertation presents a hybrid method based on high-density cross-linking/mass spectrometry and conformational space search, we review CLMS-based hybrid methods in more detail in the chapter *Related Work* (section 3.4.2).

Chapter 3

Related Work

In chapter 2, we provided a general introduction into protein structure prediction, the role of information in protein structure prediction, and hybrid methods. In this chapter, we review the literature that is closely related to the contributions of this dissertation. Our main hypothesis is that leveraging novel information sources leads to effective search in high-dimensional conformational space search. Since our work applies this principle to protein structure prediction, we mainly focus on literature from this problem domain.

Recall that the contributions of this work are: **1)** The prediction of residue-residue contacts by combining known evolutionary information with physicochemical information (EPC-map, chapters 5 and 6), **2)** leveraging high-density cross-linking/mass spectrometry data in structure determination (chapter 7), and **3)** refinement of structure constraints using corroborating information (chapter 8). Thus, we provide a review of the related work in constraint prediction with a focus on contact prediction (section 3.1), cross-link based analysis of protein structure (section 3.2), and constraint refinement strategies (section 3.3).

In addition, we present a review of algorithmic strategies to leverage tertiary structure constraints in protein modeling. This thesis also develops a contact and CLMS guided conformational space search algorithm, guided model-based search (guided-MBS, chapter 6). Thus, we focus our review on methods that use either contact or CLMS constraints in structure modeling (section 3.4).

We also review iterative or "bootstrapping" methods to structure prediction (section 3.5). MBS is an iterative algorithm and therefore a comparison with other iterative methods is appropriate. EPC-map extracts contact information from structure prediction decoys. We combine these contacts with guided-MBS. Thus, EPC-map can be viewed as a special first iteration of an iterative algorithm

3.1 Computational Prediction of Tertiary Structure Constraints

Since the size of the conformational space prevents exhaustive search [117], information about the native state is highly valuable in conformational space search [26, 243]. Thus, researchers put much effort into the prediction of native structure features with the goal of exploiting this information in protein structure prediction. As summarized in chapter 2, this includes the prediction of secondary structure [86], local tertiary structure [190], and the prediction of spatial constraints [69].

Spatial constraints are either residue-residue distances or residue-residue contacts, which are residue pairs in close, spatial proximity. Two residues are typically defined to be in contact, when the residue-residue distance of the C_β atoms is ≤ 8 Å[1].

In this section, we review methods that predict non-local spatial constraints using sequences, templates or predicted protein structures (decoys) as an information source. We structure this review by the information source that these methods employ. Since the majority of constraint prediction methods predict residue-residue contacts, the reader can assume that the discussed methods are contact prediction methods. We will specifically mark methods that predict residue-residue distances.

3.1.1 Sequence-Based Prediction of Contacts

Constraint Prediction from Sequence-Based Machine Learning

Machine learning identifies sequence patterns that are indicative of distance or contact constraints in the tertiary structure. Some researchers report the prediction of distance constraints with sequence-based machine learning [7, 109][2], but the vast majority of algorithms in this category predict contact constraints. Thus, contact prediction methods are the focus of our review.

Contact prediction methods in this category differ in machine learning algorithm, data processing, and training procedure. Many machine learning algorithms are employed for this task, such as neural networks [164, 201, 208], support vector machines [37, 226], hidden-Markov models [14], and random forests [120]. Recently, researchers also employed deep learning for contact prediction [46, 50]. These methods are typically not compared with the same experimental parameters, such as the same training and test sets. Thus, it is

[1] Note that many other contact definitions exist, such as contacts between closest side chain heavy atoms with 5 Å cutoff. Please refer to each individual publication for the exact definition.

[2] These methods predict residue-residue distances.

difficult to pinpoint the exact reasons for improvements in prediction accuracy. However, any improvement is likely a combination of novel machine learning algorithms, larger or more representative training sets, data preprocessing, and training methods.

The major advantage of sequence-based machine learning methods is that they are robust and provide meaningful predictions when only few sequences are available. Historically, these methods performed well in blind testing of *ab initio* contact prediction in the CASP experiment rounds nine and ten [141, 143]. Methods that exclusively use sequence-based machine learning were no longer superior in CASP11, but the latest deep-network instances were still among the top five methods [33].

Contact Prediction from Evolutionary Information

Evolutionary methods are exclusively used to predict residue-residue contacts. Thus, this paragraph only refers to contact prediction methods. Similar to sequence-based machine learning, evolutionary methods use sequence information to predict contacts. However, evolutionary methods heavily rely on the information inherent in multiple-sequence alignments (MSAs). Spatially close residues leave an evolutionary footprint because the spatial proximity leads to correlated mutations of the involved residues to maintain protein stability. However, simply measuring the local correlation of residue pairs leads to poor results [69], because transitive effects obfuscate correlations that are caused by direct interaction[3].

Thus, current evolutionary methods separate direct and transitive correlations by global statistical methods that infer this information from the inverse covariance matrix of the MSA. Mathematical approaches to achieve this goal are maximum entropy methods [131], graphical lasso [88], and network deconvolution [56, 198]. The most accurate methods to date use pseudolikelihood approaches to compute an approximate solution of the inverse coupling problem [53, 96]. Evolutionary methods have become increasingly popular, leading to many related algorithms that leverage evolutionary information slightly differently [93, 129, 181].

The downside of evolutionary methods is that they require many sequences to predict contacts with high precision. The number of required sequences is currently estimated to be on the order of $5L$, where L is the length of the protein chain in amino acids [96, 132]. This makes evolutionary methods ineffective if only few related sequences are available. A recent study by Kamisetty et al. [96] showed that proteins with few sequences also often do not contain structural homologs. This represents a catch-22 of evolutionary methods: These methods are ineffective for proteins that would benefit most from precise protein

[3]For example, consider three residues A, B and C, where A-C and A-B are spatially close. Because B and C are part of a (different) co-evolving residue pair, the mutations of these residues correlate, even thought the residues are not spatially close.

contacts [96]. Nevertheless, evolutionary contacts are a valuable information source for certain protein classes for which sufficient sequences are available [131].

3.1.2 Structure-Based Prediction of Contacts

In this section, we review methods that predict constraints by using structural information from either templates or predicted *ab initio* structures.

Contacts Prediction from Template Structures

Methods in this category use information from template structures that represent explicit structural information in the Protein Data Bank (PDB) [12]. Template-based approaches match the sequence profile of the query sequence and templates in the data base to find template matches [78, 158, 195, 225, 230]. Obviously, constraint prediction from accurate templates is trivial, because it only involves taking the distance information from the template. However, methods in this category use structure-derived features and machine learning to increase the accuracy of low-confidence hits, i.e. cases where a template is found, but the confidence of the database hit is low and/or the alignment contains errors [71, 98, 226]. Another line of research follows a similar principle to extract distance constraints from multiple low-confidence templates with probabilistic methods [134][4].

Even though the prediction methods operate on structural information, the retrieval of template structures is still based on sequence. Thus, these methods cannot provide constraint information for folds that do not have templates, or the templates cannot be identified. Wu and Zhang [226] showed that sequence-based methods perform better than template-based methods if templates are of low quality or unavailable. However, since these methods operate on structure input data (templates), they can, in principle, also be used for contact prediction from *ab initio* predicted protein structures.

Contact Prediction from *Ab Initio* Protein Structure Prediction

This class of methods uses information from search space samples to predict contacts. In the domain of structure prediction, these samples are candidate protein structures, also called decoys. Protein structure decoys contain physicochemical information that is encoded in energy functions. Therefore, native constraints occur more frequently in these decoys than non-native constraints.

[4] This method predicts residue-residue distances.

Methods in this category leverage the physicochemical information by analyzing the distribution of constraints to make consensus predictions [52, 176][5]. Further work in this category aims to leverage more information in the prediction process: Zhu et al. [242] added an energy-weighting to occurrence statistics of residue-residue contacts, while Blum et al. [16] used sampling statistics to predict β-strand contacts. Another approach uses sequence and structure features from low-confidence templates and *ab initio* decoys to predict contacts, which were then used to improve *ab initio* structure prediction [71, 98].

Notably, the simple heuristic of Eickholt et al. [52] resulted into the best predictor for *ab initio* contact prediction in the CASP9 experiment [143] and remained competitive in CASP10 [141].

The contact prediction approach presented in this thesis falls into this category since it uses *ab initio* decoys to predict contacts. However, our key contribution is a graph-based representation of the local contact network. We leverage the information from this representation using machine learning. As we will show later, this approach significantly improves the performance over simple consensus heuristics, similar to the approach presented in [52], in decoy-based contact prediction (chapter 5).

3.1.3 Combination Approaches to Contact Prediction

Recent research articles report improved constraint prediction by combining two or more of the aforementioned approaches.

Skwark et al. [194] and Jones et al. [89] combined sequence-based machine learning with evolutionary methods to predict protein contacts. Another body of research improves contact prediction by the integration of structural restraints. Wang and Xu [215] filtered contact maps from sequence-based machine learning with structural restraints by integer linear programming. Another group uses structure restraints as prior probabilities to refine the estimate of direct evolutionary couplings [96]. Two other approaches integrated sequence-based machine learning with low-confidence structural templates and *ab initio* models [71, 98].

These approaches improve contact prediction by combination of multiple, orthogonal information sources. We argue that this combination approach is the most accurate and widely applicable, because consensus contacts get boosted and shortcomings of one information source are compensated by other sources.

[5] The method by Samudrala et al. [176] predicted residue-residue distances.

Our contact prediction method also uses the combination of orthogonal information sources, evolutionary and physicochemical information[6], to predict contacts and maintain high prediction accuracy for a wide range of targets.

3.2 Cross-Linking/Mass Spectrometry in Protein Modeling

In chapter 2, we provided a general introduction into hybrid methods and discussed a wide range of experimental methods that can generate structural constraints for protein modeling. In chapter 7, we discuss our work on leveraging high-density cross-linking/mass spectrometry (CLMS) for structural analysis of human serum albumin (HSA). We further show that this method is able to probe HSA in its native environment, human blood serum. Therefore, we here review related work that uses CLMS for protein structure analysis. In addition, we briefly review other experimental methods with the ability to probe protein structure in natural environments.

3.2.1 Protein Structure Analysis by Cross-Linking/Mass Spectrometry

Cross-linking in combination with mass spectrometry is another approach to obtain information about tertiary structure in the form of constraints. In this approach, the experimenter incubates the protein of interest with the cross-linking reagent. This reagent is typically bi-functional and reacts to specific amino acids, usually lysines. Importantly, two residues need to be in spatial proximity in the native structure to form a link because the link can only span a certain distance, defined by the linker length. Thus a cross-link can be viewed as a "molecular ruler". The cross-linked protein is then digested by trypsin or other proteases and the resulting peptide mix is subjected to mass spectrometric analysis and database search [167, 192, 210].

Researchers mostly use CLMS constraints to model protein-protein complexes and to disambiguate templates in homology modeling.

For complex modeling, Kao et al. [97] mapped the structure of the yeast 19S proteasome with CLMS data and probabilistic modeling. Politis et al. [162] analyzed the structure proteasomal lid with CLMS and Monte Carlo optimization of the complex structure[7].

Young et al. [231] disambiguated templates of remote homologs with CLMS data to determine the correct fold of the bovine basic fibroblast growth factor.

[6] From *ab initio* structure decoys.

[7] This work was already introduced as an example to hybrid methods in chapter 2.

Other researchers report the use of CLMS data for homology and protein-protein complex modeling in as single study. Singh et al. [191] verified a monomer model of the major capsid protein E of bacteriophage lambda with CLMS data. In a second step, the authors used CLMS data to build a pseudoatomic model of the lamda procapsid shell. Chen et al. [36] performed CLMS analysis on the RNA polymerase-II-TFIIF complex, modeled the domain structures with MODELLER, and manually placed the domains to satisfy cross-link constraints.

The studies presented in this section only derived low-density CLMS data. For example, Chen et al. [36] observed 253 cross-links for a 670 kDa complex (RNA polymerase-II-TFIIF). Most studies report even lower cross-link density (compare: [97, 162, 191]). In contrast, this thesis presents a method that generates up to 1,495 cross-links for a 66 kDa protein (HSA, see chapter 7). As we will show later, high cross-link density allows the *ab initio* computation of the HSA domains.

3.2.2 Experimental Methods for Protein Structure Determination in Native Environments

Traditional approaches to protein structure impose severe restrictions to the preparation of protein sample for structural analysis. X-ray crystallography requires highly purified and crystallized protein. NMR spectroscopy can be applied to proteins in solution, but imposes limitations on protein size. This precludes structural analysis on proteins of high biological relevance, such as intrinsically unstructured proteins or long coiled-coils, membrane proteins [32], and multi-protein complexes [168]. Chromatin proteins perform their function in chromatin, which is extremely challenging to probe experimentally. An example is the methyl-CpG-binding protein 2 (MeCP2) that binds to methylated DNA in chromatin. Mutations in MeCP2 cause the autism spectrum disorder Rett syndrome [6].

Only few developments aim to overcome the restrictions of X-ray crystallography and NMR. X-ray free electron lasers only require microcrystals that are easier to obtain than regular protein crystals [21]. In-cell NMR, which modifies the NMR measurement procedure with a non-linear sampling scheme, is able to investigate the structure of small proteins in cells [173]. However, so far only few structures have been solved using this method.

Coin et al. [40] employed genetically encoded photo-activatable cross-linkers to study ligand binding to CRF Class B GPCRs. This study demonstrates that cross-linking/mass spectrometry is able to provide structural details under native conditions.

3.3 Refinement and Filtering of Constraints

In this section, we review methods that filter non-local, spatial constraints. Since we later aim to refine residue-residue contact maps and CLMS constraints with corroborating information and network analysis, we focus on previous attempts to refine this type of data or related approaches in other problem domains. Most researchers report filtering of residue-residue contact maps, which therefore is the focus of our review (section 3.3.1). To the best of our knowledge, researchers did not yet attempt network-based filtering of CLMS constraints, presumably because previous methods only provided low cross-link density (see section 3.2.1). Thus, filtering of CLMS constraints will not be part of our review. However, we will discuss related work from other domains that rely on similar concepts as the work presented in this thesis chapter 8 (section 3.3.2).

3.3.1 Filtering of Contacts

Several contact prediction algorithms filter the contact map in a final prediction step. Contact prediction algorithms usually predict contacts individually and ignore any relationship information between them. The filtering step accounts for the co-occurence likelihood of contacts.

Several researchers use stacked machine learning classifiers to filter final contact maps [46, 89, 194]. Researchers train the top classifier of the stack (filter classifier) on local contact patches[8] of varying sizes in addition to other sequence-based features (patch sizes: 11x11-15x15) [46, 89, 194]. Thus, the final classifier judges the contact probability by considering the local contact pattern surrounding that contact. This approach might contribute to the good performance of these classifiers in CASP [141, 142]. However, these methods do not leverage additional information in contact filtering other than contact probabilities and features from lower-level classifiers. In addition, these methods only consider local contact patches and therefore do not take the global connectivity of the contact map into account.

PhyCMAP filters the final contact map by constrained optimization using integer linear programming [215]. In this approach, the constraints capture physical features of real contact maps, such as the maximum number of contacts that is expected for an certain secondary structure pair and length. Thus, this approach leverages additional information, but does not use network analysis to consider global connectivity of the contact map.

Frenkel-Morgenstern et al. [62] encoded predicted contacts into a graph in which residues present nodes and contacts edges between them. They then perform network analysis (edge clustering) to filter contact maps. Since this algorithm is based on network analysis, it does

[8] Predicted from lower-level classifiers in the stack.

take the global connectivity of the contact map into account. However, Frenkel-Morgenstern et al. [62] do not use additional corroborating information to refine contacts.

In contrast to the work of Frenkel-Morgenstern et al. [62], we represent constraints as nodes and use corroborating information between constraints to represent edges. Thus, our approach combines network analysis with corroborating information (chapter 8). In the context of constraint refinement, our approach is therefore novel.

3.3.2 Corroborating Evidence and Network Analysis to Refine Noisy Data

In this section, we review two studies that use similar algorithmic concepts to our constraint refinement approach (chapter 8), albeit in different application domains.

Kamburov et al. [95] used network analysis to assess the confidence of protein-protein interactions. They transform the original interaction graph (nodes represent proteins and edges represent interactions) into a so called link graph (nodes represent interactions and edges shared interactions of the proteins). The authors then use Markov clustering to find interaction clusters. Kamburov et al. then use this cluster information to assess the protein-protein interaction confidence. This study uses a similar graph representation to the approach presented in chapter 8. However, the authors do not use any additional corroborating information.

Winter et al. [220] embedded gene expression patient data into a network of previously known relationships between genes to predict the outcome of cancer therapy. This study explicitly uses corroborating information (gene relationships) and leverages this information with the PageRank algorithm [154]. Interestingly, this approach is very similar to our approach presented in chapter 8, albeit in a different application domain.

3.4 Protein Structure Prediction with Spatial Constraints

In this section, we review studies that aim to exploit (noisy) constraint data for structure modeling. These methods convert constraint information into structural models and are therefore necessary for constraint-based structure prediction. As the focus of this thesis is *ab initio* structure prediction, we will not cover method that employ constraint data in template-based modeling. Our focus will be on structure prediction method that use spatial contact[9] data (section 3.4.1) and CLMS data (section 3.4.2).

[9]Note that some of the reviewed methods convert contact data into distances for structure prediction.

3.4.1 Using Predicted Contacts in Structure Prediction

Contact-guided structure modeling methods need to account for the noisy nature of predicted constraint data. These methods can be categorized into distance geometry methods and sampling-based methods.

Distance geometry methods transform contact constraints into a distance matrix that is then used to compute the 3D structure of the protein. This approach was originally conceived for computing structure from NMR constraints [27, 74, 180]. However, researchers also adapt this approach to model structure with predicted contacts: EV-fold uses a distance geometry algorithm to generate trial structures and refines them by molecular dynamics [131]. FT-COMAR and GDfuzz3D also employ distance geometry and differ in the transformation of contact constraints into the distance matrix [160, 205]. CONFOLD adds dihedral constraints to predicted contacts to model protein structure with distance geometry [2].

Sampling-based methods use specialized move sets and energy functions to search the native conformation with Molecular Dynamics or Monte Carlo algorithms [190]. Researchers tailor error-tolerant functional constraint energy terms to account for the noisy nature of predicted contacts [104, 138]. These studies find that constraint information, if accurate enough, allows to select native-like models from the structure ensemble. Note that similar functional contact representations have been proposed by several independent groups at approximately the same time [104, 138]. Please also note that our original study using such a contact representation (chapter 5) was also independently published in that time frame [178]. We view this as a strong indication that these functional contact representations are superior to previous functional forms in contact-guided conformational space search.

TOUCHSTONE relies on rigorous, systematic optimization of contact energy terms (among other energy terms) on a large set of high-resolution structures [237]. I-TASSER and QUARK utilize contact constraints using the same approach [236]. Wu et al. [224] optimized contact constraint weights, such that they reflect the consensus of multiple contact predictors and template hits.

MacCallum et al. [127] combined predicted contact data with physics-based molecular dynamics by converting standard energy terms and constraint energy terms into probabilities. In this framework, MacCallum et al. generates a distribution of models that is consistent with prior probabilities (physics-based energy) and posterior probabilities (likelihood of the constraint data, given a structural model). This work also explicitly acknowledges the noisy nature of predicted contact data by a per-timestep constraint sorting and selection algorithm.

3.4.2 Using CLMS Constraints in Structure Prediction

In section 2.5, we discussed the important role of methods that combine experimental data with computation. However, each experiment delivers different types of data that differ in uncertainty, ambiguity, and sparseness. Therefore, effective utilization of experimental data requires customized algorithms. In this section, we will review computational approaches to utilize experimental data from CLMS experiments. We also will use novel, high-density CLMS data in *ab initio* structure determination in chapter 7.

The use of CLMS constraints in protein structure prediction is not straightforward because of their special properties.

The C_α-C_α Euclidean distance of CLMS constraints can be very large (up to 35 Å), depending on the spacer length of the used cross-linker. Researchers use an upper C_α-C_α distance bound of 24 Å for the common cross-linking agent bis(sulfosuccinimidyl)suberate (BS3). Merkley et al. [137] validated that this distance is appropriate by the analysis of Lys-Lys distances of 766 molecular dynamics simulations. The authors recommend an even more conservative upper distance bound of 26-30 Å.

In protein-protein docking, the low spatial resolution of CLMS constraints does not impact modeling too much because long-distance CLMS constraints restrict the conformational space sufficiently to identify the interface region of the interacting proteins [97, 162].

In *ab initio* structure prediction, however, the low spatial resolution of CLMS constraints might not be sufficiently informative to guide conformational space search. Thus, researchers developed algorithms to increase the information content of CLMS data. One consideration for protein modeling is that the cross-link agent does not penetrate the folded protein but envelopes the protein along the surface, which results in tighter Euclidean distance bounds and therefore increases resolution. Xwalk follows this consideration and implements a breadth-first search on a local grid that represents the protein surface to compute the shortest solvent accessible surface distance (sSASD) [92].

Kahraman et al. [91] studied the impact of cross-links from 14 recent publications on comparative modeling, *ab initio* structure prediction, and protein-protein docking with Rosetta. The authors represent CLMS constraints *ab initio* modeling by additional energy terms and used Xwalk as a post-processing step to filter and select structural models. The authors pointed that post-processing with Xwalk was the critical step for modeling success. This indicates that their CLMS representation during search was not informative.

Unfortunately, the computational cost of sSASD is prohibitive in *ab initio* structure prediction, which requires many million energy function evaluation. Hofmann et al. [80] developed a faster calculation scheme to compute cross-links along the surface. They fit a sphere on the protein structure and compute the surface distance by the arc length between

the cross-linked residues. This study finds that this representation reduces the RMSD by 1.0 Å in *ab initio* calculations. Additionally, this study also systematically analyzes the impact of different spacer lengths as a function of protein size and the combined use of multiple cross-linking agents with different reactivities.

As we will show later, high-density CLMS data enables effective *ab initio* structure calculation, even without explicit consideration of cross-link surface distance, because the high constraint density is much more informative. Instead, we explicitly account for high noise inherent in high-density data with an error tolerant CLMS constraint representation (see section 7.3.1).

3.5 Iterative Structure Prediction Methods

Iterative structure prediction methods (or bootstrapping methods) aim to improve structure prediction by leveraging information from search space samples. These algorithms follow the key idea that new information can be discovered and utilized during search. Iterative algorithms might use constraints as input, but can also operate without input constraints. For completeness, we will discuss both cases in this section.

There are three categories of iterative algorithms. The first category maintains a population of search space samples and performs smart restart control from these samples by analyzing the search space using the sample population. Algorithms in the second category actively improve input information with new information from search space samples. The third category also improves input information, but in contrast to the second category also applies this principle to spatial information, such as contacts and distances. Iterative methods relate to two algorithms presented in this thesis: EPC-map and guided-MBS (see chapters 5 and 6). EPC-map leverages physicochemical information to actively extract information from search space samples (i.e. a single search iteration), while guided-MBS is designed to optimally leverage contact information in iterative conformational space search.

Note that some of the described algorithms fall into multiple categories and hence are mentioned multiple times throughout this section.

3.5.1 Population-Based Search Methods

Population-based search methods in structure prediction are motivated by various problem domains that are studied in computer science. An example for a population-based algorithm is the popular genetic algorithm [81]. Genetic algorithms maintain a population of candidate solutions. Each iteration, the population is modified by random modification (mutation) or

cross-over operators, such that the objective function values improve with each iteration. Another population-based search algorithm in computer science is Tabu Search [68]. Tabu Search maintains a history of search space samples in a "tabu list" that restricts search in the neighborhood of the tabu samples[10]. This avoids overconvergence into deep, local minima by enforcing search in unexplored areas of the search space.

A number of algorithms in protein structure prediction use the concept of maintaining a population of candidate conformations[11]. Conformational space annealing (CSA) maintains a "bank" of conformations and restarts search trajectories from the maintained population [116]. Raman et al. [166] implemented an iterative scheme in Rosetta that maintains a population of conformations in NMR-guided structure determination (further developed in [112] and [113]). Tyka et al. [203] adopted this concept in structure refinement. Notably, the most effective refinement algorithm in CASP11 also used a population-based approach [155]. These approaches usually need control parameters to maintain the diversity of the population and avoid early convergence into deep minima [112, 116].

Model-based search (MBS) also maintains a population of candidate conformations and restarts trajectories from the most promising samples [26]. In contrast to the methods in the previous paragraph, MBS draws additional information about the energy landscape by analyzing the distribution of conformations. MBS clusters decoys into "funnels" to build an energy landscape model. Using this model, MBS guides further search by restarting trajectories from the most promising funnels.

Replica-exchange algorithms maintain a population of conformations in several parallel trajectories (also called replicas) and exchange conformations at set intervals. Replicas differ in their Monte Carlo temperature parameter [148] or energy function [185].

I-TASSER also uses an iterative protocol. I-TASSER clusters the conformations from the first iteration and restarts TASSER simulations from the chosen cluster centroids [172].

3.5.2 Improving Input Information Using Iterative Algorithms

The next class of iterative algorithms improves input information or acquires new information during runtime. We also include algorithms that only use basic information types, such as first-principle physics. These algorithms also use search space samples to acquire information, but do not maintain an explicit population of candidate solutions. Instead, methods in this category convert search space samples into information to steer search in future iterations.

[10] Search space samples might not necessarily be explicitly represented in the tabu list.
[11] Recall that conformations are search space samples in the structure prediction domain.

This is different to population-based algorithms that use search space samples for restart control, but do not alter the sampling process or search directions[12].

Algorithms of this type are inspired by estimation of distribution algorithms. This class of algorithms builds explicit, probabilistic models of candidate solutions to steer sampling in future iterations [76, 115]. The expectation-maximization algorithm estimates a probabilistic model of a solution space that is represented by a probability density function with latent variables [47].

STAGE uses such search space samples to construct a problem-specific evaluation function [22]. Boyan [22] applied this approach to bin packing, channel routing, and learning of network structure in Bayes nets[13]. MIMIC employs a parameterized density estimator to steer search and updates the parameters with the best solutions from the previous iteration [17].

There are several algorithms in the structure prediction domain that build upon these ideas. Because of the high dimensionality of the protein structure prediction problem, classical probability density formulations are inefficient. Instead, the model is built over protein structure segments. Zipping&assembly uses molecular dynamics simulations of increasingly sized sequence chunks (starting with 8-12mers) to estimate local structure fragments. The algorithm then switches into fragment assembly to predict full length structures [153]. ItFix narrows an initial library of 3mer fragments by residue-wise estimation of secondary structure from search space samples [44]. In both cases, the algorithms use search space samples (local conformations) to acquire additional information (fragment structure) during search.

EdaFold applies an estimation of distribution algorithm to structure prediction by estimating a probability density function over the input fragments library from low-energy conformations [189]. Thus, EdaFold increases the value of the input fragments by probabilistic selection of the most promising fragments.

Blum et al. [16] used local features (such as secondary structure) from predicted structures to predict native feature values using a modified logistic regression algorithm. Blum et al. then re-picked fragments by using the predicted features. In a similar approach, Meiler and Baker [135] combined sequence-based features with structure-based features from low-energy conformations to improve secondary structure prediction using an artificial neural network. The improved secondary structure prediction leads to improved fragment selection, which in turn improves tertiary structure prediction of a second *ab initio* folding round.

[12] One could argue that search directions are altered by restarting from previous conformations. However, our view is that altering of search direction needs to modify the move set or energy function to introduce directed bias.

[13] The author considers STAGE to be a smart, multi-restart approach. However, STAGE additionally steers search by constructing the objective function, which differs from pure multi-restart approaches.

NEFILIM and RASREC explicitly construct new fragments from low-energy conformations and use the updated fragments in the next iteration [112, 186] . The constructed fragments are closer to the local backbone of the native structure and therefore improve structure prediction. Thus, these algorithms improve the input information (fragment libraries) by analyzing low-energy conformations.

3.5.3 Improving Spatial Information Using Iterative Algorithms

Algorithms in the previous section improve or acquire only local information. However, also spatial information about tertiary structure can be acquired or improved using low energy conformations.

Samudrala et al. [176] used *ab initio* structures to derive consensus distance constraints to construct protein models in a second prediction round using distance geometry. In a similar approach, I-TASSER uses predicted protein structures as probes to select structural homologs from the PDB [172]. The algorithm combines these distance constraints with constraints from threaded templates in a second TASSER round. Thus, the initial spatial information of threaded templates is improved by information from sampled conformations. Importantly, the structural homologs found with predicted structures might contain additional information that is not present in the low-energy conformation pool.

TerItFix builds upon the ItFix algorithm and obtains tertiary structure contacts from low energy conformations to bias subsequent prediction rounds towards the native structure [1].

Blum et al. [16] predicted β-sheet contacts and registers from low-energy conformations and used this information in a second resampling round.

Iterative approaches can also improve information from experimental data. Researchers usually need to manually assign resonances from NMR experiments to specific residues, which is time consuming. Meiler and Baker [136] implemented an automated spectra interpretation to obtain assignments of chemical shifts, residual dipolar couplings, and NOE constraints. This algorithm uses structural models from Rosetta *ab initio* folding simulations to optimize the assignments with Monte Carlo sampling. The algorithm then refines the best model in a feedback loop that alternatives between model building and assignment optimization. Thus, search space samples improve the quality of the initial NMR assignments.

Notably, with the exception of TerItFix, all algorithms only use one second iteration of *ab initio* structure prediction with improved information. We assume that these algorithms do not succeed to improve information after the second round because they overly exploit the information in later iterations. This prevents these algorithms to find new information in later iterations.

In this thesis, we combine our structure prediction method (EPC-map) with contact-guided model-based search (guided-MBS, see chapters 5 and 6). EPC-map represents a first iteration by broadly sampling the conformational space and extracting spatial contacts. In subsequent MBS iterations, these contacts guide model construction and sampling. Therefore, we view the combination of EPC-map and guided-MBS as a method that belongs to this category (section 3.5.3).

The work presented here contributes to iterative methods by using contact-specific physicochemical information to extract contacts from search space samples. This dissertation demonstrates that contact-specific information increases the accuracy of spatial constraints extracted from search space samples over simple heuristics similar to [52].

Chapter 4

Background

This chapter provides the technical background material that is necessary to understand the contributions of this thesis. Section 4.1 introduces the concept of network analysis. This introduction to network theory lays down the basis for our approach to encode physicochemical information with graphs. This section will also introduce centrality measures that are the basis for leveraging corroborating information with the PageRank algorithm [154]. Section 4.2 introduces the basic concepts of machine learning, classification with support vector machines, and learning from graph data. We use these concepts to learn a physicochemical model from our graph-based encoding and to predict residue–residue contacts. In section 4.3, we introduce the technical concepts of mass spectrometry, which is the central bioanalytical method to measure structural information in conjunction with chemical cross-linking. Since mass spectrometry is a wide field, we will focus on the cross-linking and mass analysis methods for reading out cross-link data (chapter 7).

4.1 Network Analysis

A network, also called graph, is a mathematical concept to represent the relationship between objects. Many fields describe object relations with networks. Thus, network analysis is an ubiquitous tool that is applied to diverse domains. Examples are the analysis of social networks [31], telecommunication networks [179], economic networks [209], the World Wide Web [156], brain networks [29], gene regulation networks [100], and protein structure networks [72].

Formally, a graph is defined by the ordered pair $G = (V, E)$, where V is a set of vertices (also called nodes) and E a set of edges that connect two vertices. Figure 4.1A shows a graph with the nodes A, B, C, D. In this graph, B has edges to all other nodes. The relation between two nodes can be asymmetric in so called directed graphs. However, we use only

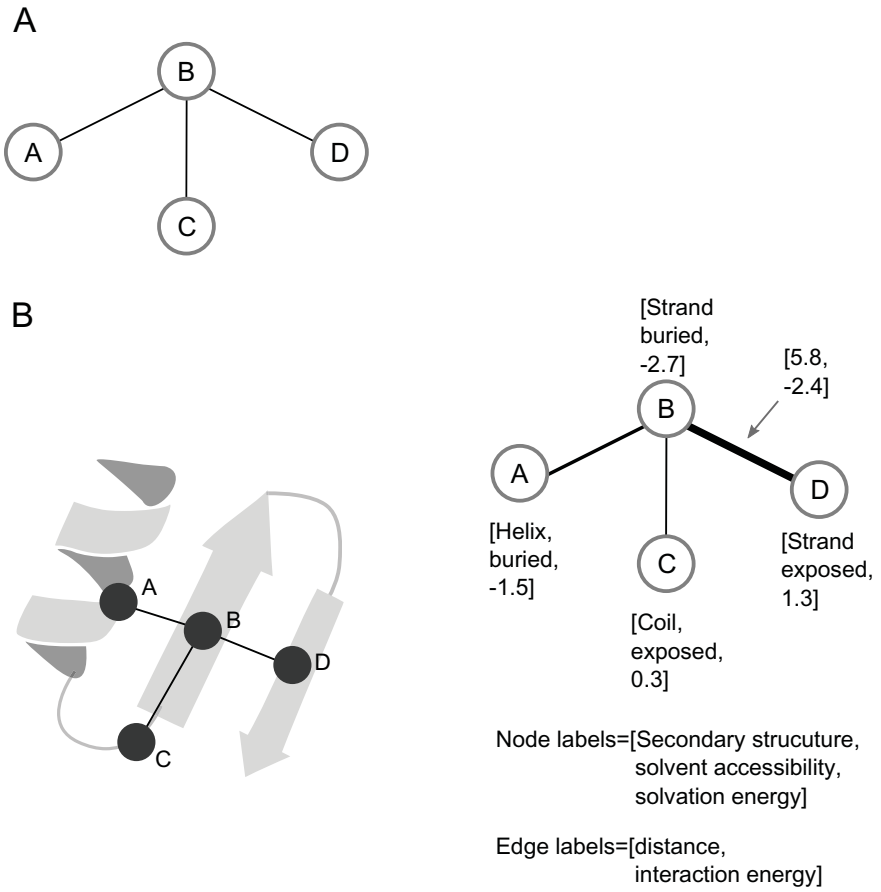

Fig. 4.1 Examples of undirected graphs. **A**: Example of an undirected graph without labels. Node B represents a hub in this network that is connected to all other nodes A, C, and D. **B**: Graph with node and edge labels to represent the properties of protein structure. The nodes represent residues and the edges represent non-covalent contacts between residues. Nodes and edges are labeled with structural and physicochemical properties.

undirected graphs in this thesis, where the edges do not indicate a direction. In addition, graph nodes and edges can encode additional properties. We exemplify this by a graph that is built from protein structure (Figure 4.1B). In this protein structure graph, nodes represent residues and edges represent non-covalent interactions between residues. Node labels additionally encode information about protein structure and physicochemistry, such as secondary structure, solvent accessibility, and residue-wise solvent energy. Edge labels encode relational information, such as Euclidean distance between residues and residue–residue interaction energies.

We employ two concepts of network analysis in this thesis. The first concept is that the network encodes information of the modeled object and that features of the graph can be used to predict this information (refer to Figure 4.2 for examples). Examples of such features

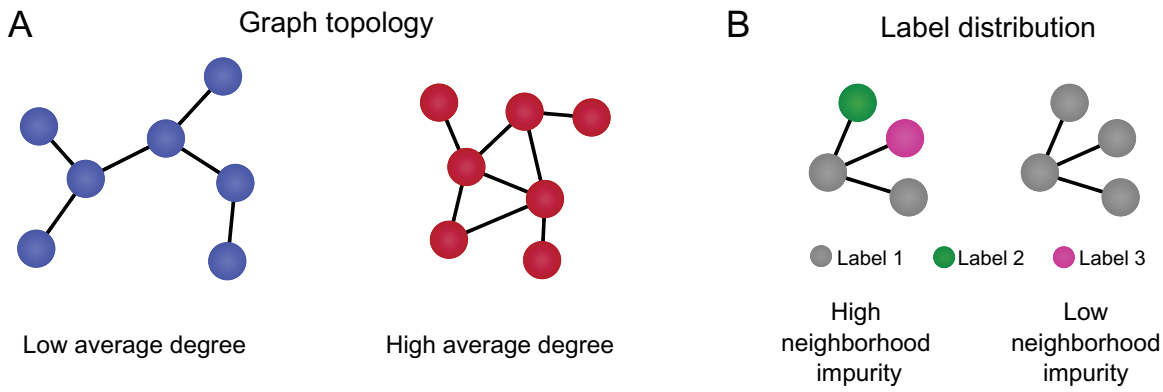

Fig. 4.2 Examples of graph features. Graph features capture topological or label distribution properties of the graph. **A**: Examples of graphs that differ in topology. Consider the average degree of the nodes in the graph. The degree of a node is defined by the number of edges that connect to that node. The blue graph has a low average degree while the red graph has a high average degree. **B**: Examples of graphs that have the same topology but differ in label distribution. In this examples, the graphs differ in their neighborhood impurity degree. For every node, the neighborhood impurity degree is equal to the number of neighboring nodes with a different label.

are topological properties of the graph as well as the distribution of node and edge labels. These features enable the derivation of a distance metric that measures the similarity of two graph objects and facilitates learning (see Figure 4.2). In this thesis, we use graph features to distinguish native from non-native contact networks to predict residue-residue contacts from physicochemical information (chapter 5).

The second concept is the notion of "hubs" in networks. Hubs are nodes that are highly connected and therefore exert strong influence on other nodes in the network. An intuitive example of hubs are web pages that are linked *from* many other web pages and link *to* many other pages. In the network of the World Wide Web, a user would encounter this page by simply following a series of hyperlinks. Thus, this page will have a high influence in this network. Additionally, this "hub" page increases the influence of all web pages it is connected to because there is a high probability that a user will land on such a page when starting from the hub page. Centrality measures quantify the importance of a node in a graph (see Figure 4.3 for an example)[1].

One special instance is the eigenvector centrality, which measures the influence of a node on all other nodes. We compute the eigenvector centrality from the adjacency matrix **A** of a graph. The adjacency matrix is a *nxn* square matrix. Each non-diagonal entry a_{ij} is the

[1]We also use centrality measures as graph features for leveraging physicochemical information (chapter 5).

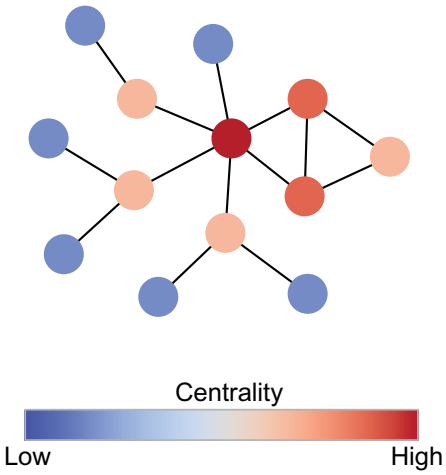

Fig. 4.3 Centrality in networks. The nodes of this network are color coded by their degree centrality. The degree centrality is the simplest centrality measure and is equal to the number of edges that connect to a node.

number of edges from node i to note j (diagonal entries are the number of edges of node i to itself). The eigenvector centrality score c_i of node n_i is defined by

$$c_i = \frac{1}{\lambda} \sum_{j \in M(n_i)} c_j = \frac{1}{\lambda} \sum_{j \in G} a_{ij} c_j$$

where $M(n_i)$ is set of neighbors of node n_i, a_{ij} is the entry for nodes i and j in the adjacency matrix, and λ is the largest eigenvector of the adjacency matrix. We use an eigenvector centrality variant (PageRank [154]) to leverage corroborating information between constraints (encoded in a graph) in chapter 8.

4.2 Machine Learning

The field of machine learning studies algorithms that learn from data to automatically improve with experience. Machine learning algorithms construct models from data to make future predictions. A common formal definition of machine learning is: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E" [139].

Researchers classify machine learning into four categories [13]:

Supervised learning: In supervised learning, the training data is labeled. These labels indicate the class or continuous response variable that the algorithm is supposed to predict [13].

Unsupervised learning: In unsupervised learning, the data is not labeled and the task of the algorithm is to discover patterns or structure in the data [13].

Semi-supervised learning: In semi-supervised learning, the data set contains unlabeled and labeled data [13].

Reinforcement learning: The algorithm learns a policy of actions from reward signals. In this learning task, the goal is to make the right decisions (actions) from this sparse signal [13].

We employ support vector machines, a supervised learning algorithm, in this dissertation. Thus, we introduce support vector machines in more detail in the next section.

4.2.1 Support Vector Machines

Support vector machines (SVM) are supervised learning algorithms for classification and regression tasks [41]. Since this thesis uses SVMs for classification, we will only introduce classification with SVMs. Intuitively, the SVM separates the training data by finding the hyperplane with the widest possible margin. Usually, many hyperplanes are possible to separate the data. However, we generally assume that unseen data of a specific category has approximately the same distribution as the training data. Under this assumption, the maximum-margin hyperplane translates into higher generalization performance (Figure 4.4A).

In many classification problems, the categories are not exactly linearly separable. Therefore, current instances of SVMs optimize a soft margin objective that allows for the misclassification of training examples [41]. For this purpose Cortes and Vapnik [41] introduced a slack variable ζ. Each ζ_i measures the degree of misclassification of a data instance x_i. An important property of SVMs are the support vectors, the subset of training instances that lie closest to the decision boundary (Figure 4.4B). Intuitively, the support vectors are the decisive data instances that, if all other training instances would be removed and training repeated, would result into the same hyperplane [30].

We now briefly summarize this mathematical derivation of support vector machines. We aim to maximize the margin $\frac{2}{||w||}$ by minimizing the distance $||w||$. For non-negative ζ_i, this is formalized by:

$$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1 - \zeta_i$$

which then results in the optimization objective

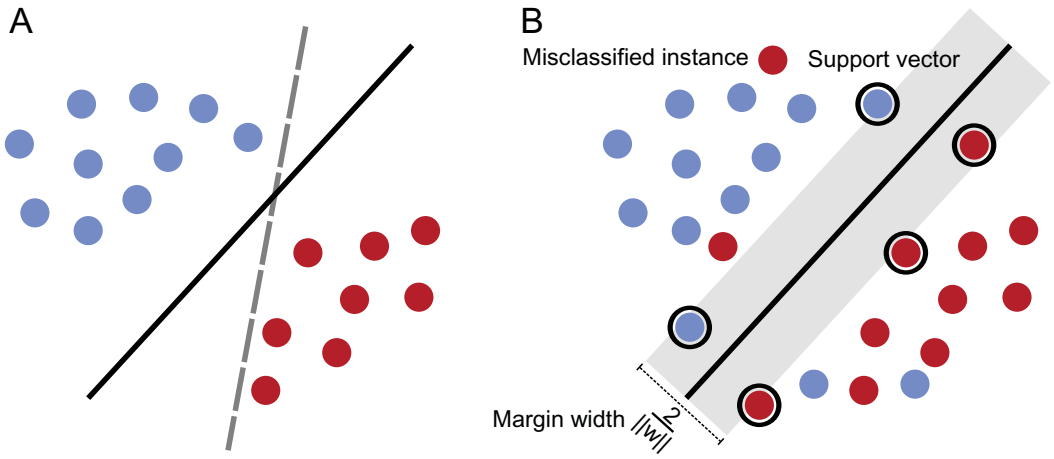

Fig. 4.4 Schematic representation of support vector machines. **A**: In general, several possible hyperplanes with different margins would separate the training data. However, we usually assume in machine learning that unseen data of a category (blue or red) will be close to the training data of the respective category. Therefore, the grey and dashed hyperplane does separate the data, but we intuitively regard the black hyperplane as a better separator, because it would generalize better under our assumption. The black hyperplane maximizes the margin to the training data, which in turn lowers the generalization error. **B**: Schematic representation of a soft-margin SVM. The soft margin formulation allows the misclassification of training instances which is measured by the slack variable. The training instances that lie within the margin are the support vectors.

$$arg\min_{\mathbf{w},\xi,b}\left\{\frac{1}{2}\|\mathbf{w}\|^2+C\sum_{i=1}^{n}\xi_i\right\}.$$

For a hinge function, the dual form optimization problem can be formalized using Lagrange multipliers. This results into the maximization objective:

$$\tilde{L}(\alpha)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j k(\mathbf{x}_i,\mathbf{x}_j)$$

for $0\leq\alpha_i\leq C$ and $\sum_{i=1}^{n}\alpha_i y_i=0$. For a full derivation of the optimization objective and a detailed introduction to SVMs, please refer to the tutorial from Burges [30].

Note that the classification of SVMs is not probabilistic but methods exist to transform the raw SVM output values into probabilities by fitting a logistic function on the output values [161].

In addition to the soft margin, Cortes and Vapnik introduced the "kernel trick" to solve the problem of inseparable categories [41]. In the kernel trick, every dot product in the SVM is replaced by a kernel function $k(\mathbf{x_i},\mathbf{x_j})$. This kernel function measures the similarity of a data

instance to all other training instances (during training) or support vectors (during prediction). Effectively, this transforms the original input into a new feature space. For a specific training instance, this new feature space is defined by the distance to all other training examples. The hyperplane is linear in the transformed kernel space, which is of higher dimensional than the input space. However, the resulting hyperplane might be non-linear in input space. Thus, the kernel trick transforms the linear SVM classifier into a non-linear classifier in input space. There exist numerous kernels for vectorized input, but the most popular is the radial basis function

$$k(\mathbf{x_i}, \mathbf{x_j}) = \exp(-\gamma \|\mathbf{x_i} - \mathbf{x_j}\|^2)$$

with the parameter $\gamma > 0$. Importantly, researchers can customize the kernel function with respect to domain knowledge. Furthermore, the kernel function can operate on non-vector training data structures, such as strings and graphs. Examples for kernels that operate on specific data structures are string kernels [122] and graph kernels [207]. We use concepts from graph kernels to learn from the graph-based representation of contact networks in chapter 5. Thus, we briefly introduce graph-based learning.

4.2.2 Learning on Graphs

As mentioned in section 4.1, graphs are natural data structures to represent the relationship between objects. However, it is not immediately clear how to perform learning on graph data with machine learning algorithms that are traditionally geared towards vectorized data. Thus, we briefly introduce learning on graph data in this section. Graph learning has important real-world applications. For example, Jie et al. [84] distinguishes schizophrenic from healthy patients with graph models from functional magnetic resonance imaging (fMRI) data. Borgwardt et al. [20] encode protein structure in a graph model to discriminate between enzymes and non-enzymes.

Learning on graphs requires the formulation of a distance metric to measure the similarity between graphs or between nodes in a graph [207]. Therefore, most research in this area deals with the derivation of distance metrics between graphs. The main challenge for deriving a graph distance metric is that the comparison between graphs is computationally expensive. Thus, graph distance metrics need to balance expressive power and efficiency.

We categorize methods to machine learning on graphs into two categories: Graph kernels and graph embedding (also called graph features). Graph kernels define kernel functions $k = (G_i, G_j)$ between pairs of graphs to directly measure their similarity. Graph kernel methods decompose the graph into parts, such as paths and trees, and compare the similarity

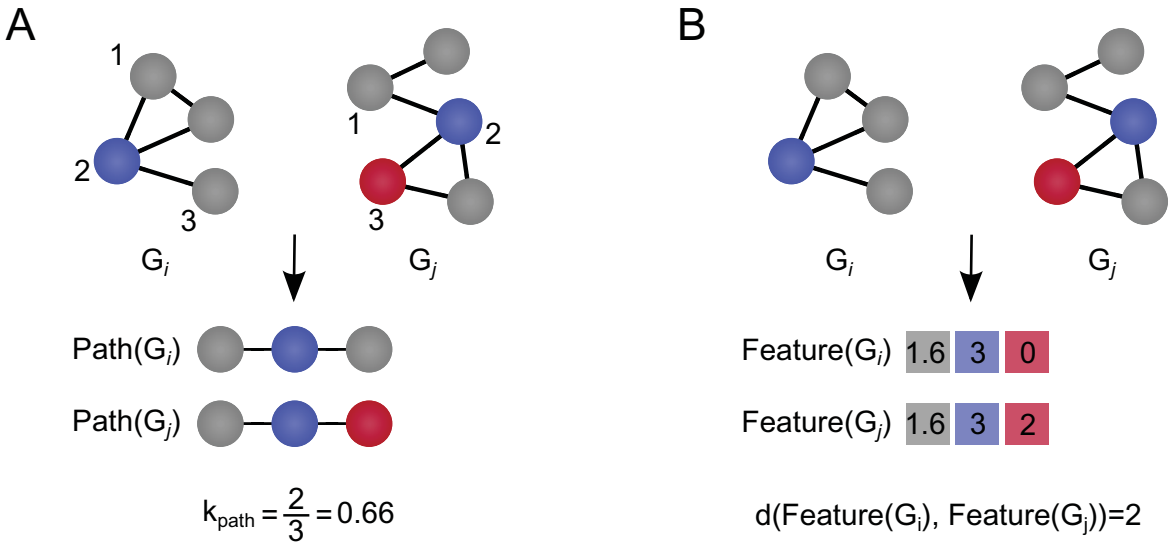

Fig. 4.5 Illustration of the graph kernel and graph feature concept. **A**: Example of a graph kernel that compares paths in graphs. We sample a random path of length three in G_i and G_j. Then, a sub-kernel k_{path} measures the similarity between these paths. In this example, the kernel simply checks whether the label (color) of a node matches along the path. The kernel repeats this process n times and sums the similarity between all paths into the final kernel value. **B**: Example of a graph feature approach to embed graphs into a vector space. In this example, the feature function measures the average node degree that coincide with each label type (blue, grey, red) and maps this value to three vector positions. The similarity between these vectors is then simply computed by Euclidean distance.

of all parts in graph G_i to all parts in graph G_j. Figure 4.5 illustrates this idea. Examples to this approach are random walk kernels, shortest path kernels, and Weisfeiler-Lehman kernels [19, 64, 184]. These approaches often incorporate domain-specific information to augment the comparison of paths [20].

The second category of methods embeds the graph into a vector space, often by extracting graph features. These approaches translate the graph data structure into a vector representation by a series of transformation functions. The functions transform topological, spectral, label distribution properties to one or multiple entries of the vector representation. Figure 4.5B illustrates this approach. Researchers use this approach in neuroimaging and in bioinformatics [84, 118].

In this dissertation, we leverage physicochemical information from contact networks by the latter type of graph learning approaches (chapter 5). The graph feature approach is computationally more efficient than graph kernels and therefore scales to larger graphs [84]. In addition, the feature engineering process is much simpler in the graph feature approach. Researchers can simply define new transformation functions to add features. On the other

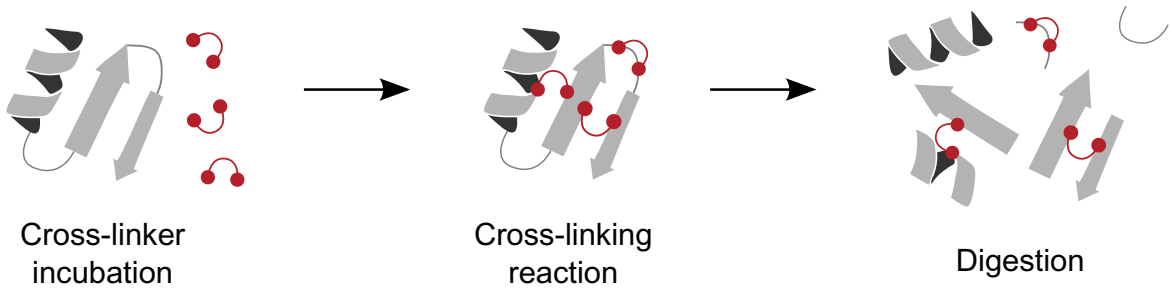

Fig. 4.6 Chemical cross-linking of proteins. In this experiment, the cross-linker incubates with the protein and chemically forms covalent bonds between residues that are within a certain distance in the native protein structure. Proteases then digest the cross-linked peptide and the resulting peptide mix is loaded on the LC-MS/MS setup.

hand, the graph feature approach can be less expressive than graph kernels because information encoded in similar paths/trees might be missed. However, in our view the simplified feature engineering process is a great benefit because feature engineering is often key to successful machine learning algorithms [48].

4.3 Cross-Linking/Mass Spectrometry

Mass spectrometry is a method to analyze the mass of atoms and molecules. The measurement process transfers the analyte to the gas phase and ionizes them. The mass spectrometer accelerates the ions in an electrical field and analyzes them in a mass analyzer that measures the mass-to-charge ratio (m/z). In this dissertation, we use mass spectrometry to obtain protein structure information from cross-linking/mass spectrometry experiments. To provide the background for these experiments, we introduce cross-linking, liquid chromatography tandem-mass spectrometry (also abbreviated as LC-MS/MS), and the identification of cross-linked residues. For more detailed introduction on this process, please refer to the review by [167].

4.3.1 Chemical Cross-Linking

In cross-linking experiments, researchers incubate the protein sample with a cross-linking reagent. The cross-linking reagent reacts with two amino acids of the folded protein (Figure 4.6). Importantly, residues can only cross-link if they are within a certain distance that is determined by the used cross-linker. Therefore, the cross-linker can be viewed as a "molecular ruler". The identification of cross-linked residues provides distance constraints that are valuable for structural modeling (see section 3.2.1 for a review of related work).

Importantly, the structural information is "stored" in the formed cross-links, which can now be subjected to analysis techniques that degrade the protein. The protein is then digested and loaded on the LC-MS/MS spectrometer.

4.3.2 Reading out Cross-Linked Peptides with LC-MS/MS

The technical challenge is the accurate identification of cross-linked peptides with mass spectrometry, which we will explain in the following sections[2]. Please refer to Figure 4.7 for an illustration of the LC-MS/MS process.

After cross-linking and digestion, the researcher loads the peptide mixture on the mass spectrometer. In a first step, a liquid chromatographic column separates the peptide mixture by hydrophobicity and results into a gradual injection of the peptides into the mass spectrometer. This procedure reduces the complexity of the peptide sample that the mass spectrometer needs to analyze. It therefore simplifies the analysis of the peptide mixture because the resulting signal is caused by a low-complexity sample (at least lower than the complexity of the non-LC separated sample).

After the LC column separated the samples, the peptides are ionized by "electrospray ionization" [57]. The peptides funnel through a needle with a tip that is held at a high electric potential. This strong electrical potential vaporizes and ionizes the peptide mixture. This process sprays the ionized peptide mixture directly into the mass spectrometer.

The mass spectrometer then determines cross-linked residues. The setup for this experiment requires at least two mass selection steps (MS/MS, also known as tandem MS or MS^2). In the first step, the mass spectrometer determines the m/z ratio of the cross-linked peptides. These "precursor ions" identify the cross-linked peptides [128, 167]. However, the first MS measurement does not reveal the exact insertion site of the cross-linker at residue resolution [167]. To increase insertion site resolution, the peptide is fragmented and then subjected to another round of mass spectrometry.

The most common fragmentation method is collision-induced dissociation [140]. An electrical potential accelerates the ions to high kinetic energy. The accelerated ions then collide with neutral gas molecules, which converts the high kinetic energy into internal energy. The high internal energy results in bond breakage, which fragments the cross-linked peptide. Note that peptides normally break along the backbone, which simplifies the linkage site analysis [167].

The second round of mass analysis identifies the fragments of the cross-linked peptide. In the best case scenario, this procedure results in complete fragmentation spectra that

[2]There exist numerous mass spectrometric setups with different ionization, fragmentation, and mass analysis methods. For the sake of brevity, we only focus on the setup that we used in this thesis for CLMS experiments.

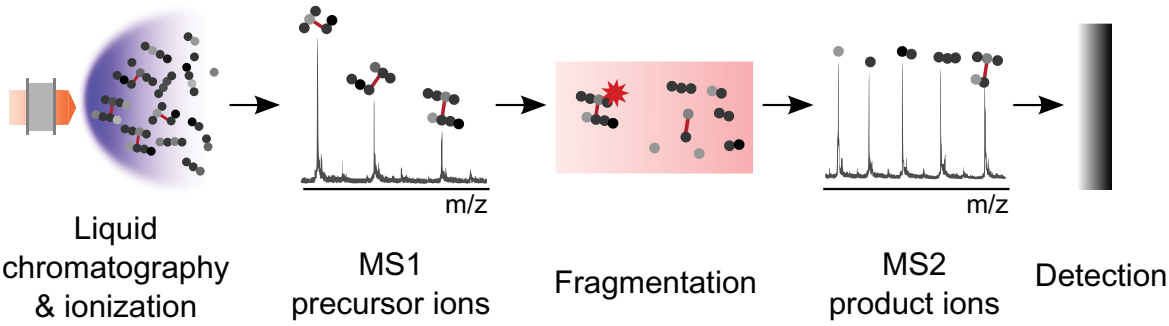

Fig. 4.7 Illustration of the LC-MS/MS setup to identify cross-linked peptides. The cross-linked peptide mix is separated by liquid chromatography, ionized, and sprayed into the mass spectrometer. In a first mass analysis step, the mass spectrometer measures the mass of the peptides (precursor ions). The next step fragments the peptides and subjects the resulting fragments (product ions) to a second round of mass spectrometry sorting. This fragmentation spectrum can identify the insertion site of a cross-link.

unambiguously identify the cross-linked residues. However, even incomplete fragmentation or missed detection of fragments can result into an approximate linkage site.

In a real-world experiment, the LC-MS/MS analysis acquires large volumes of data of the precursor and product fragmentation spectra. Identifying cross-links with high density requires the automatic processing of these spectra into residue-residue cross-link lists. We explain this process in the next section.

4.3.3 Cross-Link Identification by Database Search

The next step is the identification of cross-linked peptides (see Figure 4.8). In many aspects, the identification of cross-linked peptides is similar to the identification of linear peptides [167]. First, the algorithm selects *in silico* peptides from a protein database that approximately match the determined mass. The theoretical spectrum of this peptide is then matched to the measured fragmentation spectrum (peptide spectrum match, PSM). The resulting score measures our belief that this match is true. For cross-linked peptides, this process is more complicated because the peptide on either side needs to be matched. This leads to $(n^2+n)/2$ possible cross-links for n peptides [167]. This search problem is challenging and also increases the chances of random matches. Fischer et al. [60] further boost the confidence in peptide pair identification by considering the corroborating evidence of multiple peptide spectrum matches and folds the results further up to residue pairs (cross-links).

However, the score of each stage (peptide spectrum match, peptide-peptide pair, residue-residue pair) does not give an absolute confidence value or cutoff. Thus, the false discovery

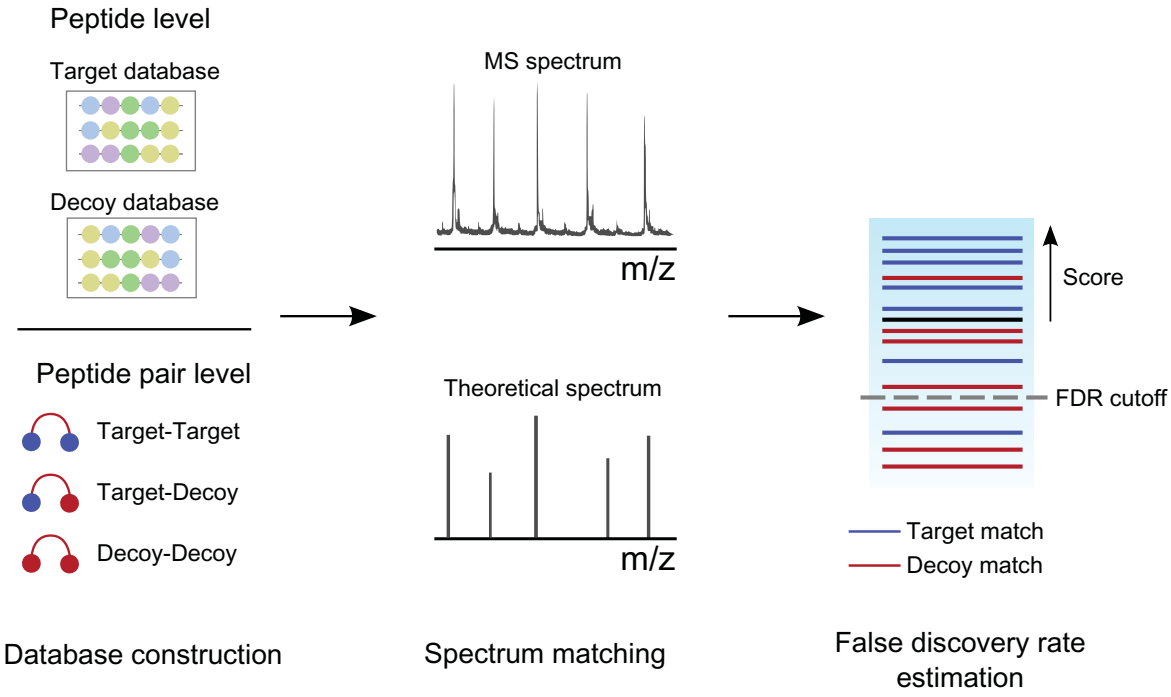

Fig. 4.8 Database search and false discovery rate estimation for cross-link identification. The target-decoy approach estimates the false discovery rate of the identified cross-links. A matching algorithm scores target and decoy (wrong sequences that do not exist in the target database) sequences by comparing the measured spectra with the theoretical spectra of the database. For peptide pairs, the decoys can consist out of target-target, target-decoy and decoy-decoy sequences. This does not provide an absolute confidence. Thus, the search algorithm matches known false positive decoys, sorts the scores of decoy and target hits, and accepts target database hits until a predefined false-discovery rate is reached.

rate (FDR) is estimated by the target-decoy[3] approach [54]. This method matches spectra against a known false positive decoy sequence database that is disjunct from the protein database. The algorithm then accepts cross-link matches until the list contains a rate of decoy matches (false discovery rate). We estimate the error for peptide spectrum matches, peptide pairs, and residue pair matches to estimate the final cross-link FDR.

Note that a true target match does not necessarily mean that the cross-linked residue pair is within the expected distance in the native structure, because the link might have formed by a different conformational state. Another error source is the possibly wrong assignment of cross-link sites caused by missing or incomplete fragmentation spectra. Thus, the number of long-distance [4] cross-links is usually higher than one would expect from the FDR.

[3]Please note that the word "decoy" has a different meaning in mass spectrometry and protein structure prediction. In mass spectrometry, decoy refers to a known false sequence, used to estimate the false discovery rate. In structure prediction, a decoy refers to a candidate predicted structural model.

[4]Longer than expected from the linker length.

Chapter 5

Combining Physicochemical and Evolutionary Information for Protein Contact Prediction

This chapter is based on the following publication:

Schneider, M. and Brock, O. (2014). Combining Physicochemical and Evolutionary Information for Protein Contact Prediction. PLoS ONE, 9(10):e108438.

Own contributions: I (MS) was the sole first author of this paper. I conceived and designed the experiments, conceived and designed the algorithm, implemented the algorithm, performed experiments, developed analysis tools, analyzed the data, and contributed to paper writing.

Contributions of co-authors: OB scientifically advised this work, conceived and designed the experiments. OB contributed to paper writing.

Extensions: This chapter extends the original manuscript by implementation details of the machine learning framework. In particular, I describe the handling of class imbalance, construction of SVM ensembles, and calibration procedure in more detail.

5.1 Introduction

The main hypothesis of this thesis is that protein structure prediction is advanced by uncovering novel information sources (see section 2.2). In this chapter, we set out to leverage an information source to predict protein contacts that received little attention in the field:

Physicochemical information from *ab initio* protein structures (see chapter 3 for related work).

Contacts contain information about the spatial proximity of two residues in the protein native structure. Several studies confirm the value of residue-residue contacts in tertiary structure reconstruction and *ab initio* structure prediction [2, 89, 98, 104, 119, 131, 138, 205, 224, 237].

Most state-of-the-art methods predict contacts using sequence information (see section 3.1). However, the performance typically deteriorates if sequence profile information is not available, which is especially true for evolutionary methods [53, 71, 88, 93, 96, 129, 131, 181]. To bypass this problem, we target a different information source that is always available in *ab initio* structure prediction: Physicochemical information in protein decoys[1].

Contacts are favorable inter-residue interactions that often result in low-energy decoys. Earlier attempts mine contact information by using energy and occurrence statistics [16, 52, 176, 242] or by machine learning [16, 71, 98] (for a detailed review of related work, see section 3.1.2 on page 20). However, energy and occurrence statistics favor all occurring contacts equally and do not distinguish between native and non-native contacts in a single decoy. Therefore, using information that is specific to native contacts should increase decoy-based contact prediction accuracy. Machine learning approaches should benefit from powerful representations that capture the information relevant for contact prediction

We contribute a method to leverage contact-specific physicochemical information by analyzing the immediate network surrounding a contact. We capture this information using a graph-based representation of the contact network. Using this representation, we train a support vector machine classifier that identifies likely native contacts in otherwise non-native structures.

We combine the physicochemical information from this algorithm with evolutionary contacts to further improve performance. Evolutionary contact prediction benefits heavily from large sequence alignments, but is not effective if sequences are not available. Physicochemical information is less susceptible by the absence of large alignments, but performs poorly if search does not identify low-energy regions in the energy landscape. Since the information sources of evolutionary and physicochemical information are largely orthogonal, the combination of these methods should maintain high prediction accuracy over a wide range of proteins. We call this combination approach EPC-map (using Evolutionary and Physicochemical information to predict Contact maps, see Figure 5.1).

EPC-map reaches 53.2% accuracy on 528 proteins for the $L/5$ top scoring long-range contacts, 7.8% higher than state-of-the-art methods (PhyCMAP and GREMLIN [96, 214]).

[1] In the context of structure prediction, a decoy is a predicted protein structure.

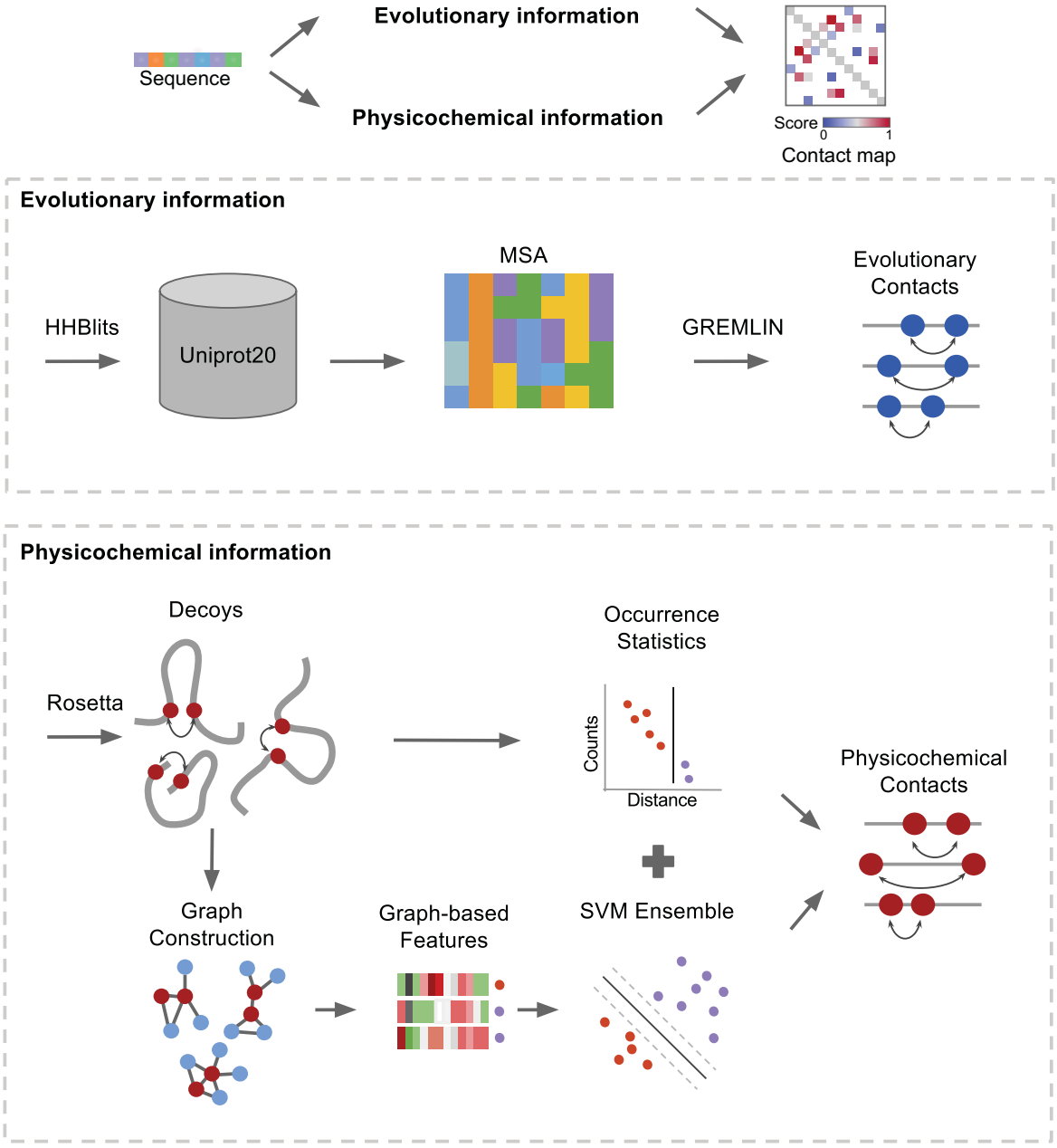

Fig. 5.1 Flowchart overview of EPC-map, combining evolutionary information (upper box) and physicochemical information (lower box). For evolutionary contact prediction, we construct multiple-sequence alignments by searching the Uniprot20 database with HHblits. GREMLIN predicts contacts from these alignments. For physicochemical contact prediction, we generate decoys with Rosetta. We then construct contact graphs from each decoy and compute the feature input vectors. An SVM ensemble predicts the contact probability from each feature vector. The SVM probability and occurrence statistics predict physicochemical contacts. Lastly, we combine evolutionary and physicochemical contact prediction to form the output of EPC-map. Figure source: Schneider and Brock [178].

We further demonstrate that EPC-map is effective, regardless how many sequences are available. This alleviates the main weakness of evolutionary methods: their high dependency on large sequence alignments. Finally, we show that EPC-map-guided Rosetta calculations improve the GDT_TS of 132 proteins by 7.8 over unguided Rosetta.

Section 5.2 provides a general overview of the algorithm. In section 5.3 we present a detailed description of the implementation of EPC-map. The results and discussion follows in section 5.4. Section 5.5 concludes this chapter.

5.2 Overview of the Algorithm

We first provide an overview of our approach. Please refer to Figure 5.1 for a flowchart overview of EPC-map. EPC-map is divided into three components:

1. Prediction of contacts from evolutionary information
2. Prediction of contacts from physicochemical information
3. Combination of contacts from evolutionary and physicochemical information

5.2.1 Prediction of Contacts from Evolutionary Information

EPC-map implements a standard pipeline for the prediction of evolutionary contacts. The algorithm searches for sequence homologs in the UniProt sequence database [202] using HHblits [169]. The resulting multiple-sequence alignment (MSA) forms the input to GREMLIN which predicts evolutionary contacts with a pseudolikelihood model [96].

5.2.2 Prediction of Contacts from Physicochemical Information

For the prediction of physicochemical contacts, EPC-map first generates decoys with Rosetta [170]. The independent trajectories of Rosetta lead to a broad sampling of the search space. In the next steps, EPC-map mines contact information from low-energy decoys by constructing contact graphs.

Contact Graphs

We use undirected graphs to model the neighborhood of a contact (see Figure 5.2). Nodes in this graph represent residues and edges represent contacts between the residues. We label the nodes and edges with physicochemical, structural, and evolutionary characteristics. A complete description of these labels can be found in the appendix (Appendix B).

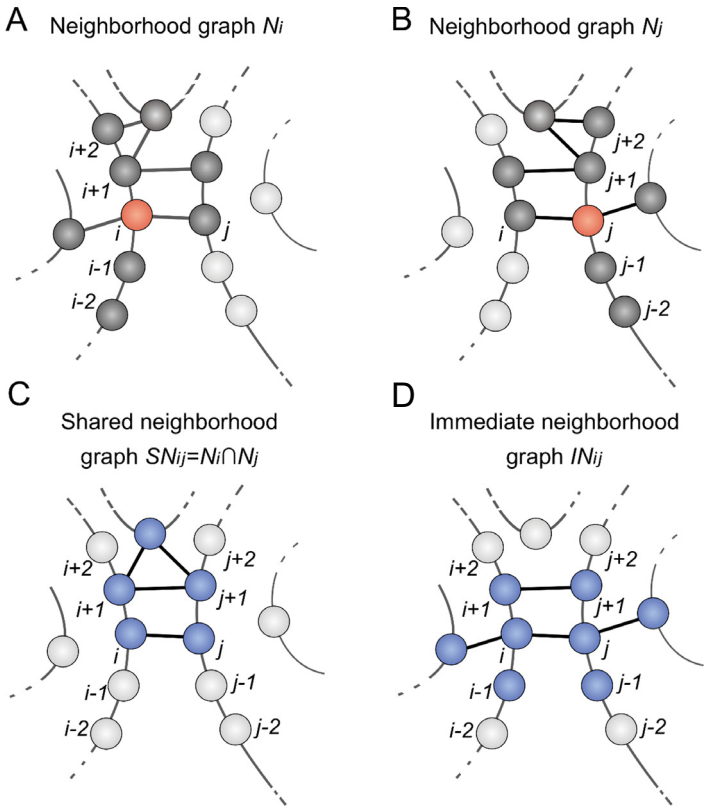

Fig. 5.2 Definition of graphs used to model the neighborhood of the contacting residues i and j: Nodes represent residues (circles), edges represent contacts (solid black lines). **A**: The neighborhood graph N_i for residue i contains all residues in contact with residues $i-2, i-1, i, i+1$, and $i+2$ (dark grey). **B**: The neighborhood graph N_j. **C**: The shared neighborhood graph SN_{ij} for the contact between residues i and j is defined by the intersection of N_i and N_j. Residues that belong to SN_{ij} are shown in blue. Shared neighborhood graphs capture the local context of the shared neighborhood of the contacting residues. **D**: The immediate neighborhood graph IN_{ij} is defined by all residues that are in contact to i or j. Residues that belong to IN_{ij} are shown in blue. Immediate neighborhood graphs capture the direct neighborhood of the contacting residues. Figure source: Schneider and Brock [178].

We first define the neighborhood of residue i as all residues within two positions in sequence (which corresponds to residues $i-2, i-1, i, i+1, i+2$). The neighborhood is extended by all residues that are also in contact to those. Residues in an α-helix have the same facing on subsequent helix turns. Thus, we use the $i-4$, i, $i+4$ residues to define the neighborhood of residues in α-helices. This captures the residues that point into the same direction in the helix interface. This defines the neighborhood graph N_i for residue i (see Figure 5.2A and B).

Neighborhood graphs are the basic unit to define the local context of a contact C_{ij} between residue i and j. We construct two types of contact graphs that capture a different

context. The shared neighborhood of residues i and j is defined by the intersection of the neighborhood graphs N_i and N_j. Formally, the shared neighborhood graph (SN_{ij}) is defined by $SN_{ij} = N_i \cap N_j$. Thus, the shared neighborhood graph includes all residues that are in contact with residues i and j and with their sequential neighbors (Figure 5.2C). The immediate neighborhood graph (IN_{ij}) captures the direct influence of residues i and j on the contact network. This graph includes all residues that are in direct contact with residue i or j.

These graphs are the fundamental data structure of EPC-map. We then convert these graphs into graph-based features that are the input to the support vector machine. Next, we provide an overview of these features.

Overview of Features for Contact Prediction from Physicochemical Information

We convert the network characteristics of contacts in decoys into $n = 48$ features. Each feature corresponds to one or several binary or real-values. We convert the shared neighborhood graph and immediate neighborhood graph independently into features. Therefore, the final input vector will contain each graph feature twice. The input vector is the single vector of all concatenated inputs. The support vector machine will use these vectors for training and prediction.

We group features into seven categories: pairwise, graph topology, graph spectrum, single node, node label statistics, edge label statistics, and whole protein features. Table 5.1 summarizes these feature categories.

Table 5.1 Overview of the features used for contact prediction. A detailed description of the features is given in Appendix B. Table source: Schneider and Brock [178].

Group	Feature examples	Number of inputs
Pairwise	Chemical type, secondary structure, solvent accessibility, sequence separation, hydrogen bonding, sequence separation from N/C-terminus, contact potential, distance, average distance in ensemble, mutual information	49
Graph topology[a]	Number of nodes, number of edges, average degree centrality, average closeness centrality, average betweenness centrality, graph radius, graph diameter, average eccentricity, number of end points, average clustering coefficient	10
Graph spectrum[a]	Largest two eigenvalues, number of different eigenvalues, sum of eigenvalues, energy of adjacency matrix	5
Single node[a]	Degree, closeness centrality, betweenness centrality, sequence conservation and sequence neighborhood conservation for i and j	10
Node label statistics[a]	Chemical type of residues, secondary structure descriptors, solvent accessibility, hydrogen bonding, average free solvation energy, 4-bin solvation energy distribution, entropy of labels, neighborhood impurity degree, average distance from centroid, sequence conservation, sequence neighborhood conservation	43
Edge label statistics[a]	Link impurity, 5-bin mutual information distribution, cumulative mutual information, 3-bin contact potential distribution	12
Whole protein	Amino acid composition, secondary structure composition, length class	29

[a] Graph-based features

We now provide the motivation for each of the feature categories.

Pairwise features capture the chemical type, secondary structure, and solvent accessibility of the amino acids i and j.

Topological graph features characterize the contact network. These feature captures physicochemical characteristics, such as residue packing. Nodes in densely packed regions have a higher degree in the contact network than those in loosely packed regions. Dense packing is considered to be a feature of native protein folds. Thus, contacts in densely packed regions are more likely to be native. This can be measured by network descriptors, such as the average degree centrality of the graph.

Spectrum graph features reflect the connectivity of the graph and are based on the eigenvalues of the adjacency matrix. Similar to topological features, they provide another way to capture physicochemical characteristics of the contact network.

Single node features capture the topological properties of the contacting residues i and j. We calculate these features separately for residues i and j.

Node and edge label statistics complement topological features by encoding additional information that is not captured by topological or spectrum features that are invariant to node and edge labels. For example, the folding of globular proteins is driven by the hydrophobic effect [197]. Thus, well-folded structures should have a protein core that consists of hydrophobic residues. The chemical type distribution (encoded by node labels) of residues captures this property.

Whole protein features capture global information of the protein, such as the composition of amino acids, secondary structure, and chain length.

We give a detailed description of each feature in Appendix B.

5.2.3 Combination of Contacts from Evolutionary and Physicochemical Information

In a last step, EPC-map combines evolutionary and physicochemical contacts using linear combination to form the final prediction output.

5.3 Implementation

In this section, we describe the implementation of EPC-map in more detail. This will include parameters of various parts of the EPC-map pipeline, as well as version numbers of software and databases. Section 5.3.1 contains details about the generation of multiple-sequence alignments and section 5.3.2 about the prediction of evolutionary contacts.

5.3.1 Generation of Multiple-Sequence Alignments

EPC-map generates multiple-sequence alignments by searching a clustered UniProt [202] database with HHblits [169]. EPC-map uses multiple-sequence alignments to predict evolutionary contacts and to compute local sequence features in contact graphs, such as residue conservation.

EPC-map runs HHblits (version 2.0.11) with default parameters and uses a UniProt sequence database (dated March 2013) with a maximum pairwise sequence similarity of 20%.

5.3.2 Evolutionary Contact Information

We predict evolutionary contacts using GREMLIN with the multiple-sequence alignment from HHblits as input. The authors of GREMLIN provided us with a version of GREMLIN (without version number) that we run with default parameters.

5.3.3 Decoy Generation

We first generate protein decoys that are the source of physicochemical information. We generate decoy with the `AbinitioRelax` protocol of Rosetta version 3.2 [170]. This protocol implements a low-resolution prediction phase and a high-resolution energy minimization phase. In the low-resolution phase, the protocol assembles the structure from short 3mer and 9mer fragments with a reduced representation of the protein chain. In this representation, the side-chains are not explicitly modeled. Instead, they are replaced with a virtual "centroid" atom. The energy function of the low-resolution phase is a knowledge-based force field. In the high-resolution phase, Rosetta adds side-chains and performs energy minimization of the decoy with a hybrid physics-based/knowledge-based all-atom potential

We now describe the generation of decoys that are used for training of our algorithm, and the generation of decoys that are used during prediction.

Decoys for Training

Ab initio structure prediction generates decoys along the entire spectrum of quality, from close to the native to far away from the native structure (as measured by backbone RMSD or GDT_TS). Our goal for training EPC-map is to generate decoys that span this entire spectrum for as many proteins as possible. We implement this by using three independent Rosetta runs that differ in their degree of native bias [203]

We introduce the native bias by three different fragment libraries[2]. The fragment libraries differ in their fragment quality. The first fragment library excludes any homologs to the target sequence. This results in decoys that are of low quality and quite far away from the native structure (min/max/mean/median GDT_TS of 10.7/73.5/26.5/24.3 for all proteins. The second fragment library contains fragments from homologous proteins. Because fragments from homologs have backbone conformations that are closer to the target protein, decoys computed with this fragment library are closer to the native structure (min/max/mean/median GDT_TS of 11.2/99.3/28.1/25.4). In the third fragment library, we take fragments from the native structure itself. Sampling with this fragment library generates decoys that are even closer to the native structure than the other two libraries (min/max/mean/median GDT_TS of 10.4/99.8/37.0/31.2).

Note that we only use these native biases for training to generate decoys with a wide range of quality. The native bias is not used during prediction, making EPC-map an *ab initio* method.

We generate 200 decoys with each fragment library, 600 decoys per protein in total. We retain the decoys with 3% lowest energy from each set.

Decoys for Prediction

We generate 1000 decoys without homologous fragments for contact prediction. Thus, the fragments in the prediction case do not contain native bias. We save the top 2% decoys, as judged by Rosetta all-atom energy. The absence of native bias is important to ensure a method that is effective in *ab initio* structure prediction. Note that EPC-map does not use any structural information from homologous structures (templates). This differentiates EPC-map from methods that utilized threaded templates for contact prediction

[2]We quantify the impact of the native bias by the GDT_TS of the five lowest-energy decoys for each protein. Two completely dissimilar structures have a GDT_TS of 0, while a perfect structural match results in an GDT_TS of 100.

5.3.4 Overview of Software for Feature Generation and Machine Learning

The key innovation of our algorithm is the graph-based representation of the local contact context and the features that we extract from these graphs. For an overview of the features, please refer to section 5.2.2 and for a detailed description of all features to Appendix B.

Here, we list all external software that we employ to compute graph-based features. In addition, we list all publications that introduced features that we use in EPC-map, or our features are inspired from. We compute solvent accessibility and free solvation energy with POPS [34]. We use STRIDE [63] to compute secondary structure content in decoys and assign hydrogen bonding. EPC-map implements residue-wise sequence conservation measures that are described by Fischer et al. [58]. Cheng and Baldi [37] introduced several pairwise feature that we use in EPC-map. We employ a contact potential from a contact prediction study that uses random forests and sequence features to predict contacts [120].

The NetworkX Python package handles all graph-based operations in EPC-map [75]. SVM training and prediction steps employ the scikit-learn [157] library, which implements a Python wrapper of LIBSVM [35].

Finally, Li et al. [118] demonstrated the effectiveness of many topological and spectrum features that we use in this study.

In the next section, we give implementation details on how we perform training and prediction with support vector machines on these features.

5.3.5 Training of SVMs with Physicochemical Information

The training of SVM models for contact prediction from physicochemical information contains three key components: Handling of class imbalance, construction of SVM ensembles, and calibration of the SVM output.

Handling of Class Imbalance

Imbalanced learning problems are a serious issue in machine learning because the performance of many machine learning algorithms deteriorates in the case of class imbalance. However, many real world problems are inherently imbalanced. For example, the construction of classifiers for medicine is challenging because the availability of "positive" samples (patient has a disease) is usually much lower than for "negative" samples (patient does not have a disease). Thus, the machine learning community intensively researched the issue

of class imbalance and suggested a number of algorithms that address the class imbalance problem (He and Garcia [77] wrote a comprehensive review on that topic).

The contact prediction problem is also imbalanced because there are much more non-native contacts than native contacts in the decoys of our training set (see section 5.3.3 for computation of training decoys).

We handle the imbalanced learning problem of contact prediction with random undersampling, which is a common technique to address the imbalance problem [77]. In this approach, we tune the ratio of contact and non-contact examples in response of prediction accuracy. We empirically find that a 1:3 (contacts/non-contacts) ratio of training example maximizes the prediction accuracy. For training, we sample our training instances for both the contact class and the non-contact class. For each protein, we perform random undersampling by randomly selecting 50 native and 150 non-native contacts. We find that increasing the number of training instances does not significantly improve prediction accuracy but increases training and prediction time.

Construction of SVM Ensembles

The downside of random undersampling is the loss of information because many training instances are not regarded for learning. In addition, the resulting SVM can be biased to a specific set of training instances, which can lead to high variance. We train an ensemble of SVM models to compensate this effect to some degree. Each SVM in the ensemble trains on a disjunct set of proteins and performs its own random undersampling. We first split the proteins in the training set into five disjunct subsets. We train a SVM classifier with each set by randomly sampling 50 native and 150 non-native contacts (see section 5.3.5). This procedure reduces the impact of information loss by using multiple SVM instances. In addition, using an SVM ensemble in such a "bagging" approach is shown to reduce variance and therefore increases generalization performance [204].

We use an ensemble of five SVMs, one for each protein subset. We sample approximately 30,000 training instances for each subset. We normalize each individual input column in the input data by subtracting each input by its mean and then dividing it by standard deviation.

SVM training and hyperparameter tuning: We use the radial basis function kernel for training and prediction. This results in two hyperparameters that we need to tune. The cost controls the penalty of misclassified training instances. The kernel parameter γ controls the width of the radial basis function kernel. We determine the cost and γ parameter by 10-fold cross-validation on the EPC-map_train data set (introduced later in section 5.3.9)

using grid search. Our optimization objective is the long-range $L/5$ accuracy. We obtain best performance for $c = 10$ and $\gamma = 0.001$.

Calibration of SVM Output

The output of an SVM is the decision function value that reflects the distance of a query instance to the hyperplane. Since we train each individual SVM with a disjunct set of training instances, each SVM will fit a hyperplane into its training subset. As a result, the decision function values of the SVM instances might not be comparable for a particular query instance i.e. a query might have very different decision function values.

We calibrate each SVM instance to cope with this issue and make the SVM outputs comparable. This is critical for the combination of SVM instances in a bagged ensemble. We obtain calibrated probability estimates by a binning procedure [232].

First, we compute all decision function values for each SVM on hold out training instances. For this purpose, we use the training instances from all proteins that are not in the training set of the SVM. Note that for calibration, we use all calibration examples and do not perform undersampling.

We group the raw SVM decision function values between the 5 and 95 percentile into ten bins. For each bin, we compute the probability of a native contact. For prediction, the calibrated probability values form the output of each SVM.

Our experiments suggest that this procedure is more effective than Platt's method [161] to improve prediction accuracy for contact prediction from physicochemical information.

5.3.6 Prediction of Contacts from Physicochemical Information with SVMs

EPC-map performs contact prediction for a protein by considering the 2% lowest-energy Rosetta decoys (see section 5.3.3), constructing contact graphs (see section 5.2.2), and converting this contact graphs into features that capture the physicochemical properties (see section 5.2.2).

EPC-map than scores each contact in each decoy using the SVM ensemble. The probability $p(C_{ij})$ for single contact C_{ij} in one decoy is given by:

$$p(C_{ij}) = \frac{1}{l} \sum_{k=1}^{l} p^k_{\text{SVM}}(C_{ij}),$$

where $p^k_{\text{SVM}}(C_{ij})$ denotes the probability output value of the k-th SVM.

The sample contact C_{ij} might appear multiple times in different decoys. Thus, we compute the final score $S_{\text{ENS},ij}$ of a contact C_{ij} by averaging the SVM probability $p(C_{ij})$ over all decoys in the *decoy ensemble*:

$$S_{\text{ENS},ij} = \frac{1}{m}\sum_{n=1}^{m} p(C_{ij}^n).$$

In this equation, $S_{\text{ENS},ij}$ denotes the score of the contact between residues residues i and j. C_{ij}^n is the contact C_{ij} in the n-th decoy, $p(C_{ij}^n)$ is the SVM ensemble output of C_{ij} in the n-th decoy. The number of decoys containing the contact is denoted by m.

5.3.7 Combination of Evolutionary and Physicochemical Information

Finally, EPC-map combines evolutionary, occurrence statistics, and physicochemical information. We combine the output of the SVM system with the frequency f_{ij} of contact C_{ij} in the decoy ensemble. Then, we combine the resulting score with the GREMLIN output $S_{\text{GREMLIN},ij}$:

$$S_{\text{EPC-map}}(C_{ij}) = \beta(\alpha f_{ij} + (1-\alpha)S_{\text{ENS},ij}) + (1-\beta)S_{\text{GREMLIN},ij}.$$

We tune the α and β parameters by optimizing the $L/5$ accuracy of long-range contacts by five-fold cross-validation on the training set.

Depending on the number of available sequences, the output values from GREMLIN scale differently. Because GREMLIN computes contacts from multiple-sequence alignments, its performance is highly dependent on the number of available sequences. If $5L$ or more sequences are available, GREMLIN performs well in template discrimination tasks [96]. This indicates a $5L$ sequences performance threshold for GREMLIN.

Since we know the number of available sequences at prediction time, we can construct a system that supplement GREMLIN's predictions when many sequences are available and compensates GREMLIN's loss in accuracy when sequence availability is limited. Thus, we tune two separate sets of β parameters for proteins with $< 5L$ sequences and for proteins with $\geq 5L$ sequences.

We obtain the following parameter values by this procedure: $\alpha = 0.425$ and $\beta = 0.275$ for proteins with $< 5L$ and $\beta = 0.35$ for proteins with $\geq 5L$ sequences, respectively.

The final output of EPC-map is the list of contacts, ordered by their score.

5.3.8 Using Contact Constraints in *Ab Initio* Structure Prediction

In this thesis, we also evaluate the utility of EPC-map contact maps as distance constraints. This adds additional terms to the Rosetta energy function that scores the agreement of a

decoy with the predicted contacts. Critically, predicted contacts represent an uncertain source of constraints because they contain false positives. Thus, we devise an energy term that maximizes the number of satisfied contacts for a given conformation, instead of penalizing violated constraints. This is important, because a penalization of contacts could lead to unfavorable energies if only a small number of constraints is violated.

We incorporate contact constrains with a modified Lorentz function L into the energy function of Rosetta:

$$L(d_{ij}) = \begin{cases} -\frac{1}{\pi}\frac{\frac{1}{2}w}{(d_{ij}-l)^2+(\frac{1}{2}w)^2} & if \quad d_{ij} < l \\ -\frac{1}{\pi}\frac{\frac{1}{2}w}{(\frac{1}{2}w)^2} & if \quad l < d_{ij} \leq u \\ -\frac{1}{\pi}\frac{\frac{1}{2}w}{(d_{ij}-u)^2+(\frac{1}{2}w)^2} & if \quad u < d_{ij} \end{cases}$$

where d_{ij} is the distance between residues i and j in the decoy.

The function contains further parameters, such as the lower bound l, the upper bound u and the half width w. We use l = 1.5 Å, u = 8 Å and w=1.0. The half width parameter w controls the decrease in energy when d_{ij} is not within the lower/upper bounds, with w being the half-width, i.e. the violation where $E_{max}/2$ is still rewarded (E_{max} is the maximum energy bonus).

There are three cases in which a particular distance d_{ij} in a decoy interacts with the modified Lorentz function:

1. **Case 1**: If $l < d_{ij} < u$, the maximum energy bonus E_{max} is added to the energy
2. **Case 2**: If $d_{ij} - l \leq 2w$ or $u - d_{ij} \leq 2w$, the constraint is only mildly violated and a decreased energy bonus is still rewarded.
3. **Case 3**: If $l - d_{ij} > 2w$ or $d_{ij} - u > 2w$, the constraint is significantly violated and the energy bonus falls back to zero i.e. the constraint is not penalized and ignored.

The former case ($l - d_{ij} > 2w$) does not happen when modeling constraints with l = 1.5 Å and w=1.0. However, this case is meaningful for other cases (for example l=10 and u=20) and allows application of the modified Lorentz function in other cases (such as modeling CLMS constraints).

5.3.9 Data Sets

In this section, we describe the construction of training and validation data sets for EPC-map. EPC-map_train is the training set and EPC-map_test, D329, SVMCON_test, CASP9-10_hard, CASP10 and CASP10_hard are the validation sets in our study.

EPC-map_train: For the construction of the training set, we first culled the PDB using PISCES [211]. We used the following search parameters: Chain length of 50–150 amino acids; at most 25% sequence identity; 0–2 Å resolution for X-ray structures. We constructed the test set from smaller proteins to speed up method development since algorithmic design cycles can be tested much faster with small proteins.

We filtered the PISCES set by the following criteria: *a)* Protein chains that contain chain breaks. Li et al. [120] defines a chain break by a distance of 4.2 Å or larger between C_α atoms of two residues adjacent in sequence. *b)* Chains with extended structures and chains with structures that are significantly affected by packing against other chains or by large interior bound ligands.

Additionally, we filtered the protein chains for structural redundancy. We perform pairwise alignment with Deepalign [214] and remove chains with GDT_TS of 60 or more to any other chain in the training set[3].

In a last step, we filtered the training set with proteins from the validation sets. This is required to ensure that the training set is disjunct from any of the validation sets. This enables an unbiased assessment of the performance of EPC-map. For this final filtering, we removed all chains from EPC-map_train with more than 25% sequence identity or a GDT_TS > 60 to any chain in EPC-map_test, D329, SVMCON_test and CASP9-10_hard.

We randomly remove 15% of the remaining chains to construct EPC-map_test. After this procedure, EPC-map_train consists of 742 chains. Appendix C lists the PDB IDs of this training set.

EPC-map_test: We randomly select 132 proteins from the training set to form this set. We do not use proteins in this set for training or parameter tuning. Appendix C lists the PDB IDs of this test set.

D329: The D329 data set [120] consists of 329 chains with 55–458 amino acids.

SVMCON_test: The SVMCON_test data set contains 48 medium-sized protein chains (46–198 amino acids) [37]. Protein 1aaoA is listed as a theoretical model in the PDB and is therefore removed.

CASP9-10_hard: In addition to the test sets from literature, we constructed a test set that reflects the performance of EPC-map at the CASP experiment. Proteins in the CASP contact prediction category are usually "hard" proteins without known templates (free modeling, FM)

[3]We remove only chains if the aligned region covers more than 60% of the smaller protein.

or templates that are difficult to identify. We only considered chains that are solely comprised by free modeling domains. This also excludes all proteins that contain at least on TBM or TBM_hard domain. This is necessary to assess the performance on hard modeling targets, since prediction performance of targets with TBM and FM domains could only stem from accurate contact prediction in the TBM domain.

The resulting proteins are difficult tertiary structure prediction targets because they do not have templates, many sequence homologs, or have unusual folds. Thus, these proteins represent cases for which contact prediction is most useful.

This procedure results in 16 protein chains from the CASP9 experiment and four protein chains from the CASP10 experiment (20 total).

CASP10: We also performed contact prediction experiments on all proteins from CASP10. We used 104 proteins from the CASP website which had crystal structures available at the time of this study.

CASP10_hard: Finally, we evaluate our approach on difficult protein targets from CASP10. Unfortunately, there are only four proteins in CASP10 that exclusively contain FM or TBM/FM domains. Thus, we relaxed the stringent requirement for difficult proteins to chains that contain at least one FM or TBM/FM domain. In total, 14 proteins fulfill this criterion. This is less stringent than the "pure FM" criterion in the CASP9-10_hard set, but nevertheless provides some indication of the performance of EPC-map on difficult protein targets.

The inclusion of the CASP10 and CASP10_hard sets enables us to compare EPC-map with all CASP10 contact prediction groups. We downloaded the results from the CASP10 groups from the CASP website http://www.predictioncenter.org. Note that we perform all predictions on the CASP10/CASP10_hard data set with a re-trained version of EPC-map that only used databases and proteins that are released before CASP10 (May 2012). This results in EPC-map having the same information available as all other methods in CASP10, namely information dated May 2012 or earlier. This ensures a fair comparison of EPC-map and all other CASP10 methods.

5.4 Results and Discussion

We structure the analysis of EPC-map in three parts: Performance evaluation of EPC-map, analysis of EPC-map under different conditions, and the impact of EPC-map predictions on *ab initio* protein folding.

First, we evaluate the performance of EPC-map on the six test data sets. We then discuss how sequence alignment depth and protein chain length improves performance of EPC-map. We also analyze the SVM ensemble and their contribution on prediction performance. Finally, we analyze the improvement of *ab initio* structure prediction by EPC-map contact maps.

Unless otherwise stated in this chapter, we focus our discussion on long-range contacts, since they are of most value in structure modeling [177]. We measure contact prediction performance by the accuracy and coverage of the $L/10$, $L/5$ and $L/2$ top scoring contacts, with L being the length of the protein chain.

5.4.1 Comparison of EPC-map Performance with Top CASP10 Methods

Table 5.2 Contact prediction performance of several methods on the CASP10 data set (104 proteins). Table source: Schneider and Brock [178].

Method	Range	Acc[a](SE[b])/Cov[c][L/10]	Acc[a](SE[b])/Cov[c][L/5]	Acc[a](SE[b])/Cov[c][L/2]
EPC-map	Long	0.561(0.030)/0.064	0.492(0.028)/0.105	0.378(0.024)/0.188
IGB-Team (CMAPpro)	Long	0.338(0.027)/0.033	0.285(0.023)/0.057	0.208(0.017)/0.100
MULTICOM-construct (DNcon)	Long	0.327(0.025)/0.030	0.285(0.021)/0.053	0.215(0.015)/0.101
SAM-T08	Long	0.288(0.024)/0.028	0.260(0.020)/0.050	0.202(0.016)/0.093
ProC_S4	Long	0.285(0.026)/0.030	0.245(0.022)/0.048	0.183(0.015)/0.091
RaptorX-Roll	Long	0.308(0.024)/0.032	0.269(0.020)/0.053	0.211(0.016)/0.097
MULTICOM-novel (NNcon)	Long	0.212(0.021)/0.020	0.167(0.017)/0.031	0.120(0.011)/0.055
Counting	Long	0.338(0.030)/0.039	0.272(0.025)/0.059	0.184(0.016)/0.096
GREMLIN	Long	0.498(0.031)/0.047	0.448(0.029)/0.082	0.341(0.025)/0.153
PSICOV	Long	0.447(0.031)/0.036	0.375(0.028)/0.061	0.284(0.023)/0.117
PhyCMAP	Long	0.365(0.026)/0.031	0.325(0.022)/0.059	0.246(0.016)/0.106
EPC-map	Medium	0.648(0.026)/0.159	0.537(0.025)/0.258	0.362(0.020)/0.406
IGB-Team (CMAPpro)	Medium	0.429(0.026)/0.105	0.361(0.022)/0.173	0.263(0.015)/0.301
MULTICOM-construct (DNcon)	Medium	0.454(0.026)/0.106	0.377(0.021)/0.177	0.276(0.016)/0.313
SAM-T08	Medium	0.381(0.021)/0.090	0.322(0.017)/0.153	0.237(0.013)/0.270
ProC_S4	Medium	0.447(0.024)/0.106	0.370(0.020)/0.176	0.272(0.014)/0.316
RaptorX-Roll	Medium	0.464(0.022)/0.109	0.393(0.019)/0.181	0.301(0.015)/0.300
MULTICOM-novel (NNcon)	Medium	0.400(0.027)/0.093	0.329(0.022)/0.148	0.248(0.016)/0.257
Counting	Medium	0.543(0.031)/0.121	0.453(0.027)/0.201	0.308(0.019)/0.326
GREMLIN	Medium	0.473(0.029)/0.110	0.380(0.026)/0.171	0.242(0.019)/0.256
PSICOV	Medium	0.429(0.030)/0.088	0.345(0.026)/0.137	0.229(0.019)/0.215
PhyCMAP	Medium	0.474(0.023)/0.103	0.418(0.021)/0.179	0.309(0.016)/0.321

[a] Accuracy
[b] Standard error
[c] Coverage

Methods that perform well in CASP experiment are considered to be the state-of-the-art. Thus, we first evaluate EPC-map on proteins of the CASP10 experiment. This enables us to compare the performance of our method to several methods that participated in CASP10. We downloaded the results of other methods from the CASP10 website. For this assessment,

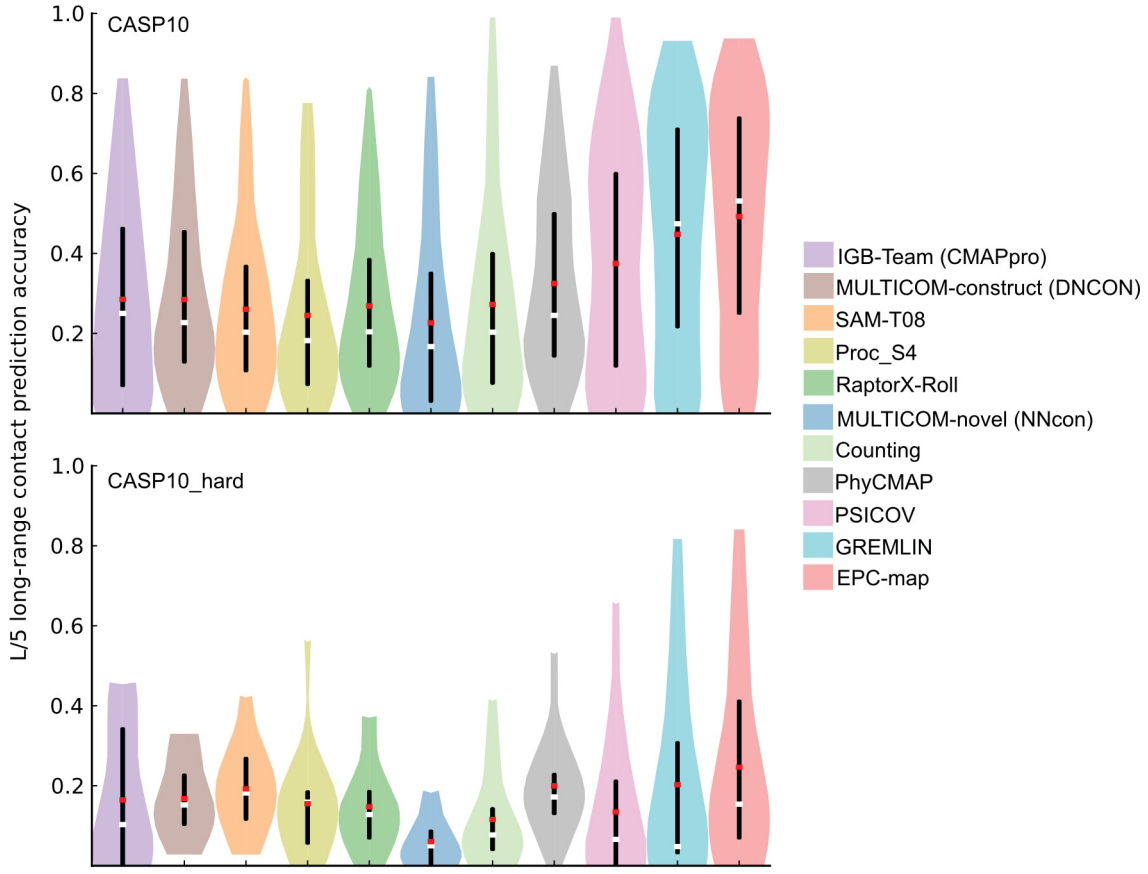

Fig. 5.3 Prediction performance overview for the CASP10 and CASP10_hard data sets. The figure shows the long-range contact prediction performance of the top scoring L/5 contacts. Different methods are shown as color coded violin plots. The lower and upper end of the black vertical bars in each violin denote the accuracy at the 25 and 75 percentile, respectively. White horizontal bars indicate the median, red horizontal bars the mean accuracy. The distribution of the prediction accuracies for individual proteins is indicated by the shape of the violin. Figure source: Schneider and Brock [178].

we do not use information from structural homologs in decoy generation. This allows us to compare EPC-map with the performance of other *ab initio* sequence-based methods.

We compare EPC-map with the top six methods in CASP10 [141] that submitted predictions for all targets. We excluded, to the best of our knowledge, all methods that rely on templates or use server models to predict protein contacts.

This selection results in the following six methods:

Group 305 (server name: IGB-Team, program name: CMAPpro) [46], Group 222 (server name: MULTICOM-construct, program name: DNCON) [50], Group 358 (server name

Table 5.3 Contact prediction performance of several methods on the CASP10_hard data set (14 proteins). Table source: Schneider and Brock [178].

Method	Range	Acc[a](SE[b])/Cov[c][L/10]	Acc[a](SE[b])/Cov[c][L/5]	Acc[a](SE[b])/Cov[c][L/2]
EPC-map	Long	0.280(0.081)/0.026	0.246(0.068)/0.046	0.169(0.045)/0.076
IGB-Team (CMAPpro)	Long	0.201(0.062)/0.015	0.165(0.047)/0.025	0.126(0.032)/0.049
MULTICOM-construct (DNcon)	Long	0.192(0.036)/0.015	0.168(0.027)/0.028	0.130(0.018)/0.054
SAM-T08	Long	0.222(0.041)/0.018	0.192(0.032)/0.033	0.147(0.024)/0.062
ProC_S4	Long	0.170(0.050)/0.013	0.155(0.037)/0.025	0.115(0.022)/0.047
RaptorX-Roll	Long	0.130(0.033)/0.011	0.147(0.029)/0.027	0.130(0.025)/0.060
MULTICOM-novel (NNcon)	Long	0.102(0.029)/0.008	0.074(0.018)/0.012	0.051(0.011)/0.020
Counting	Long	0.148(0.041)/0.013	0.116(0.033)/0.022	0.084(0.019)/0.039
GREMLIN	Long	0.257(0.082)/0.022	0.203(0.068)/0.034	0.155(0.050)/0.064
PSICOV	Long	0.191(0.071)/0.016	0.135(0.052)/0.022	0.104(0.036)/0.042
PhyCMAP	Long	0.240(0.049)/0.020	0.200(0.034)/0.033	0.154(0.026)/0.064
EPC-map	Medium	0.438(0.089)/0.110	0.344(0.074)/0.171	0.238(0.055)/0.263
IGB-Team (CMAPpro)	Medium	0.317(0.079)/0.085	0.258(0.060)/0.136	0.189(0.038)/0.229
MULTICOM-construct (DNcon)	Medium	0.348(0.068)/0.079	0.279(0.053)/0.121	0.216(0.041)/0.237
SAM-T08	Medium	0.300(0.061)/0.074	0.286(0.048)/0.147	0.203(0.039)/0.239
ProC_S4	Medium	0.286(0.054)/0.070	0.242(0.042)/0.117	0.185(0.030)/0.220
RaptorX-Roll	Medium	0.342(0.075)/0.082	0.283(0.057)/0.135	0.222(0.040)/0.233
MULTICOM-novel (NNcon)	Medium	0.278(0.070)/0.067	0.232(0.051)/0.108	0.176(0.034)/0.192
Counting	Medium	0.360(0.083)/0.088	0.311(0.064)/0.151	0.211(0.041)/0.239
GREMLIN	Medium	0.270(0.084)/0.060	0.213(0.064)/0.090	0.156(0.056)/0.137
PSICOV	Medium	0.277(0.091)/0.061	0.216(0.078)/0.085	0.156(0.062)/0.136
PhyCMAP	Medium	0.346(0.077)/0.077	0.303(0.062)/0.136	0.232(0.041)/0.249

[a] Accuracy
[b] Standard error
[c] Coverage

RaptorX-Roll), Group 113 (server name: SAM-T08 Server, program name: SAM-T08) [101], Group 314 (server name: Proc_S4), and Group 424 (MULTICOM-Novel, program name: NNcon) [201].

We categorize these six methods into four classes. Neural network systems: (SAM-T08 and MULTICOM-Novel), deep neural network systems (IGB-Team and MULTICOM-construct), random forest methods (Proc_S4), and methods that implement a novel context-specific distance-based statistical potential [240].

We also include PhyCMAP in the analysis that is shown to outperform current methods on the CASP10 data set [214]. PhyCMAP uses a sequence-based random forests to predict a coarse contact map and enforces physical constraints using linear integer programming to make the contact maps more "protein like". We obtain predictions from PhyCMAP by its web service. In addition, we include PSICOV [88] and GREMLIN [96] in our analysis. Both methods predict contacts from evolutionary information. We run locally installed versions of these programs. Finally, we evaluate contact prediction using only occurrence statistics in protein decoys (we refer to this as Counting).

Note that some of the top prediction methods from CASP9 and CASP10 use such a occurrence statistic heuristic [141, 143]. However, methods in CASP9/CASP10 collect decoys from other CASP servers that utilize numerous different scoring functions, templates, or structure generation methods. In contrast, the Counting approach in this thesis uses only decoys that are generated by Rosetta.

The long-range $L/5$ contact prediction performance of all methods on the CASP10 data set is summarized in Figure 5.3. Detailed medium and long-range contact prediction performance is shown in Tables 5.2 and 5.3.

The contact prediction accuracy of EPC-map is 0.492 on the CASP10 data set. GREMLIN is the second-best method with a mean accuracy of 0.448. PhyCMAP ranks third with a mean accuracy of 0.325. The top performing method in the CASP10 experiment is MULTICOM-construct(DNCON) [141]. During CASP10, MULTICOM-construct(DNCON) reached 0.285 main accuracy on the CASP10 data set. In summary, EPC-map predicts contacts higher accuracy than GREMLIN (+4.4%) and MULTICOM-construct(DNCON) (+20.7%) on the CASP10 data set.

The majority of CASP10 proteins can be modeled with templates. However, contact prediction is most useful for structures that lack templates because templates usually provide enough information to compute a good structure model. Thus, we repeat the analysis on CASP10 proteins that contain free modeling or difficult free-modeling/template-based domains .

Overall, the prediction accuracy on this difficult data set is much lower than on the entire CASP10 set. The top performing CASP10 methods reach 0.165-0.192 mean accuracy. From the remaining methods, GREMLIN and PhyCMAP have the best mean accuracy of 0.203 and 0.200, respectively. Contacts by EPC-map have an average accuracy of 0.246 on this set. Therefore, EPC-map is more accurate than GREMLIN (+4.3%) and the top performing CASP10 method (+5.4%) on (CASP10_hard).

5.4.2 Performance of EPC-map on Data Sets from Literature

The most rigorous assessment of EPC-map performance would be the comparison with the best methods of the CASP10 experiments on all data sets. However, the web services of the best-performing groups only allow the input of a single sequence (and process only one sequence at the time) and are therefore not suited for the processing of hundreds of proteins. Additionally, standalone versions of the CASP10 contact prediction methods were not available at the time of this study.

In lieu of re-implementing these methods or manually using several servers that only allow input of single sequences, we focus on the evaluation of methods that are implemented

in standalone versions or web services that are designed for batch processing. Methods that fulfill these criteria are: NNcon, PhyCMAP, Counting, PSICOV and GREMLIN.

Comparison against these methods is justified because PhyCMAP and GREMLIN perform as good or better than the best method in CASP10 (see Figure 5.3). Therefore, we consider these methods to represent the state of the art. Therefore, the comparison with these methods is a fair estimate of state-of-the-art performance[4].

Table 5.4 Contact prediction performance of EPC-map, Counting, GREMLIN, PSICOV, PhyCMAP and NNcon on the CASP9-10_hard data set (20 proteins). Table source: Schneider and Brock [178].

Method	Range	Acc[a](SE[b])/Cov[c][L/10]	Acc[a](SE[b])/Cov[c][L/5]	Acc[a](SE[b])/Cov[c][L/2]
EPC-map	Long	0.414(0.064)/0.046	0.322(0.049)/0.072	0.222(0.031)/0.122
Counting	Long	0.246(0.055)/0.028	0.176(0.034)/0.043	0.120(0.017)/0.072
GREMLIN	Long	0.230(0.062)/0.022	0.193(0.053)/0.038	0.134(0.038)/0.063
PSICOV	Long	0.192(0.054)/0.018	0.157(0.048)/0.030	0.111(0.034)/0.052
PhyCMAP	Long	0.277(0.044)/0.029	0.225(0.034)/0.046	0.169(0.023)/0.088
NNcon	Long	0.097(0.031)/0.008	0.089(0.021)/0.016	0.080(0.021)/0.041
EPC-map	Medium	0.445(0.076)/0.146	0.343(0.053)/0.223	0.216(0.033)/0.332
Counting	Medium	0.407(0.075)/0.138	0.312(0.059)/0.200	0.196(0.036)/0.301
GREMLIN	Medium	0.166(0.038)/0.055	0.120(0.027)/0.081	0.080(0.015)/0.140
PSICOV	Medium	0.150(0.040)/0.043	0.114(0.033)/0.063	0.086(0.027)/0.105
PhyCMAP	Medium	0.311(0.055)/0.087	0.273(0.047)/0.153	0.186(0.027)/0.269
NNcon	Medium	0.158(0.042)/0.049	0.141(0.034)/0.085	0.122(0.026)/0.173

[a] Accuracy
[b] Standard error
[c] Coverage

We show the long-range $L/5$ contact prediction accuracy for the remaining data sets (CASP9-10_hard, EPC-map_test, D329, SVMCON_test) in Figure 5.4. Tables 5.4–5.7 contain a detailed analysis of the performance on these data sets.

Available Sequence Distribution Determines the Difficulty to Predict Contacts

The test data sets in our study differ significantly in their distribution of available sequences in the MSA (see Figure 5.5). We argue that the number of sequences is a good measure for the difficulty to predict correct contacts of a data set. This is confirmed by studies on contact prediction with evolutionary methods [96, 132]. In the analysis of contact prediction of CASP10, Monastyrskyy et al. [141] did not find a link between alignment depth and contact prediction accuracy. However, the authors of this study constructed the alignments

[4] We use a version of EPC-map that is trained on sequences and structures dated after CASP10. This best reflects the performance of our method, given the current sequence and structure information.

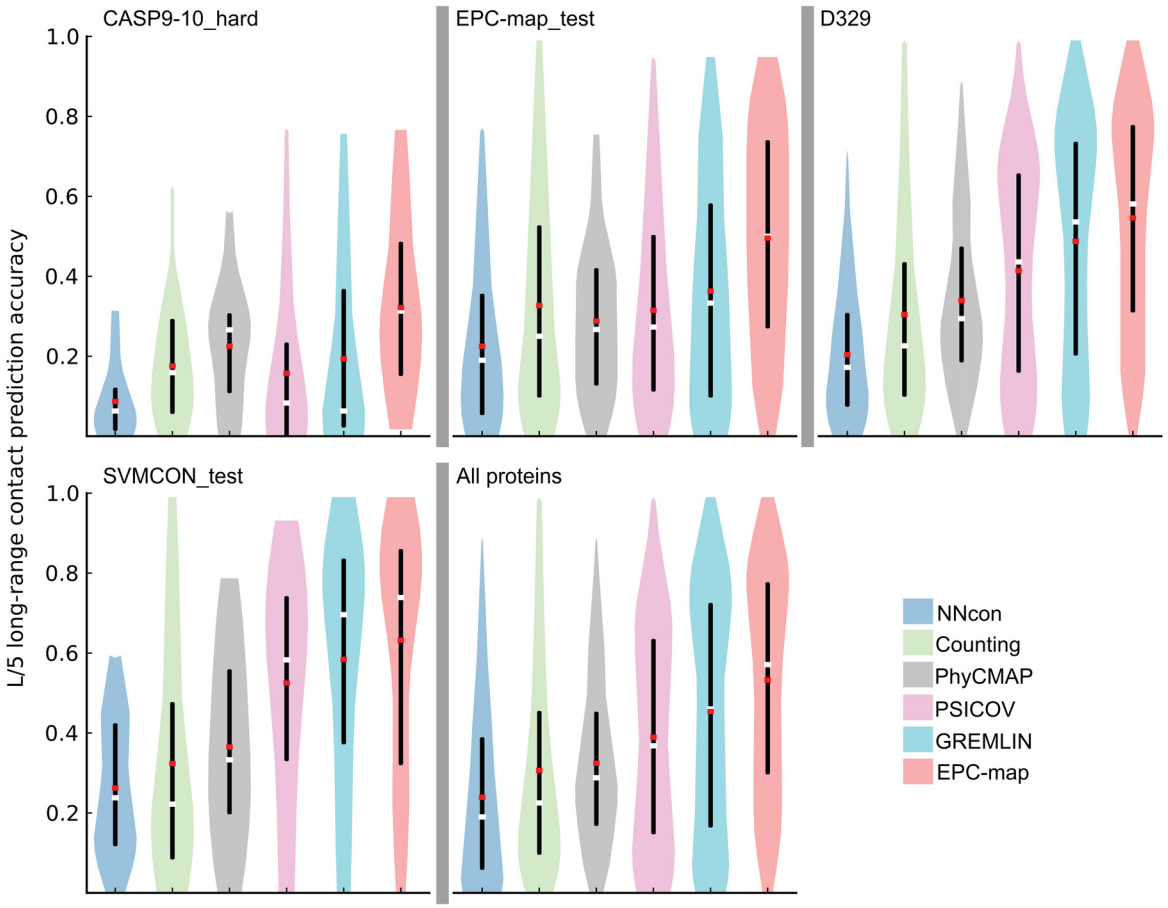

Fig. 5.4 Prediction performance overview for the CASP9-10_hard, EPC-map_test, D329 and SVMCON_test data sets. The figure shows the long-range contact prediction performance of the top scoring L/5 contacts. Different methods are shown as color coded violin plots. The lower and upper end of the black vertical bars in each violin denote the accuracy at the 25 and 75 percentile, respectively. White horizontal bars indicate the median, red horizontal bars the mean accuracy. The distribution of the prediction accuracies for individual proteins is indicated by the shape of the violin. Data sets are sorted from difficult (CASP9-10_hard) to easy (SVMCON_test). The last panel shows the pooled results for all proteins from these data sets. Figure source: Schneider and Brock [178].

themselves with PSI-BLAST [5]. In contrast, predictors employed more powerful methods in CASP10 to search for sequence homologs, such as HHblits [169] or JackHMMER [85]. Thus, the alignment depth in this analysis might not reflect the actual number of sequences the predictors had available. We will show later that the number of available sequences affects all contact prediction algorithms (section 5.4.3). This argument is further supported by the sophisticated homologs sequence search engine that is employed by the winning contact prediction method in CASP11 [105].

Table 5.5 Contact prediction performance of EPC-map, Counting, GREMLIN, PSICOV, PhyCMAP and NNcon on the EPC-map_test data set (132 proteins). Table source: Schneider and Brock [178].

Method	Range	Acc[a](SE[b])/Cov[c][L/10]	Acc[a](SE[b])/Cov[c][L/5]	Acc[a](SE[b])/Cov[c][L/2]
EPC-map	Long	0.553(0.026)/0.059	0.496(0.023)/0.109	0.376(0.019)/0.205
Counting	Long	0.378(0.028)/0.042	0.327(0.024)/0.076	0.263(0.018)/0.150
GREMLIN	Long	0.426(0.027)/0.044	0.363(0.024)/0.077	0.268(0.019)/0.139
PSICOV	Long	0.383(0.026)/0.040	0.315(0.021)/0.066	0.218(0.016)/0.114
PhyCMAP	Long	0.320(0.020)/0.033	0.288(0.016)/0.061	0.228(0.012)/0.123
NNcon	Long	0.251(0.022)/0.024	0.225(0.017)/0.045	0.183(0.012)/0.095
EPC-map	Medium	0.632(0.023)/0.159	0.523(0.021)/0.264	0.358(0.016)/0.437
Counting	Medium	0.577(0.026)/0.147	0.475(0.023)/0.245	0.322(0.016)/0.400
GREMLIN	Medium	0.408(0.025)/0.100	0.313(0.021)/0.152	0.197(0.013)/0.234
PSICOV	Medium	0.339(0.024)/0.082	0.226(0.018)/0.128	0.173(0.012)/0.203
PhyCMAP	Medium	0.440(0.023)/0.106	0.363(0.018)/0.178	0.273(0.013)/0.329
NNcon	Medium	0.335(0.024)/0.079	0.296(0.020)/0.138	0.209(0.013)/0.246

[a] Accuracy
[b] Standard error
[c] Coverage

Table 5.6 Contact prediction performance of EPC-map, Counting, GREMLIN, PSICOV, PhyCMAP and NNcon on the D329 data set (329 proteins). Table source: Schneider and Brock [178].

Method	Range	Acc[a](SE[b])/Cov[c][L/10]	Acc[a](SE[b])/Cov[c][L/5]	Acc[a](SE[b])/Cov[c][L/2]
EPC-map	Long	0.613(0.016)/0.057	0.546(0.015)/0.102	0.421(0.013)/0.195
Counting	Long	0.363(0.017)/0.036	0.304(0.014)/0.061	0.219(0.010)/0.107
GREMLIN	Long	0.545(0.017)/0.047	0.487(0.016)/0.086	0.368(0.013)/0.162
PSICOV	Long	0.485(0.017)/0.041	0.414(0.015)/0.072	0.293(0.011)/0.128
PhyCMAP	Long	0.393(0.014)/0.033	0.339(0.011)/0.059	0.256(0.008)/0.110
NNcon	Long	0.236(0.011)/0.020	0.204(0.009)/0.035	0.156(0.006)/0.067
DNCON[d]	Long	-	0.329(0.037)/0.066	-
EPC-map	Medium	0.663(0.014)/0.173	0.563(0.013)/0.287	0.380(0.011)/0.464
Counting	Medium	0.578(0.016)/0.148	0.476(0.014)/0.241	0.323(0.010)/0.395
GREMLIN	Medium	0.468(0.017)/0.115	0.369(0.014)/0.179	0.230(0.009)/0.274
PSICOV	Medium	0.411(0.016)/0.100	0.320(0.013)/0.152	0.202(0.008)/0.237
PhyCMAP	Medium	0.471(0.013)/0.113	0.406(0.011)/0.196	0.295(0.008)/0.356
NNcon	Medium	0.380(0.015)/0.087	0.324(0.012)/0.149	0.231(0.008)/0.266
DNCON[d]	Long	-	0.427(0.036)/0.192	-

[a] Accuracy
[b] Standard error
[c] Coverage
[d] Values reported in the original paper of DNCON [50]

Table 5.7 Contact prediction performance of EPC-map, Counting, GREMLIN, PSICOV, PhyCMAP and NNcon on the SVMCON_test data set (47 proteins). Table source: Schneider and Brock [178].

Method	Range	Acc[a](SE[b])/Cov[c][L/10]	Acc[a](SE[b])/Cov[c][L/5]	Acc[a](SE[b])/Cov[c][L/2]
EPC-map	Long	0.679(0.044)/0.067	0.632(0.044)/0.127	0.482(0.036)/0.233
Counting	Long	0.394(0.048)/0.042	0.323(0.042)/0.072	0.238(0.027)/0.131
GREMLIN	Long	0.642(0.048)/0.059	0.584(0.045)/0.113	0.455(0.038)/0.215
PSICOV	Long	0.609(0.047)/0.057	0.525(0.042)/0.100	0.360(0.031)/0.167
PhyCMAP	Long	0.414(0.039)/0.039	0.373(0.034)/0.073	0.270(0.023)/0.127
NNcon	Long	0.297(0.031)/0.027	0.263(0.025)/0.050	0.197(0.017)/0.094
DNCON[d]	Long	-	0.326(0.011)/0.052	-
EPC-map	Medium	0.714(0.032)/0.172	0.589(0.031)/0.283	0.405(0.028)/0.455
Counting	Medium	0.558(0.038)/0.136	0.475(0.032)/0.227	0.334(0.026)/0.380
GREMLIN	Medium	0.547(0.045)/0.128	0.450(0.040)/0.212	0.279(0.027)/0.314
PSICOV	Medium	0.501(0.046)/0.111	0.375(0.037)/0.168	0.243(0.025)/0.275
PhyCMAP	Medium	0.490(0.035)/0.110	0.407(0.029)/0.181	0.307(0.022)/0.337
NNcon	Medium	0.354(0.038)/0.078	0.314(0.032)/0.136	0.265(0.023)/0.272
DNCON[d]	Long	-	0.368(0.011)/0.190	-

[a] Accuracy
[b] Standard error
[c] Coverage
[d] Values reported in the original paper of DNCON [50]

Under this argumentation that alignment depth affects contact prediction performance, the number of available sequences is a measure of the difficulty of a data set. CASP9-10_hard is the most difficult data set with more than 70% of proteins having between 1 and 3.2L sequences in the MSA. The easiest data set is SVMCON_test with more than 60% of proteins having 10L or more sequences in their alignment.

Performance on CASP9-10_hard

EPC-map has a mean accuracy of 0.322 for the difficult CASP9-10_hard and improves over the next best method (+9.7%, see Figure 5.4). Counting and GREMLIN are on opposite ends of the spectrum of approaches to contact prediction. Counting is a pure structure-based approach while GREMLIN only uses evolutionary information. Interestingly, neither of these methods has good mean accuracy on the CASP9-10_hard (Counting: 0.173; GREMLIN: 0.193). EPC-map uses a combination approach that results in much higher performance. This is an indication that evolutionary and physicochemical information are orthogonal information sources and their combination leads to more accurate contact maps.

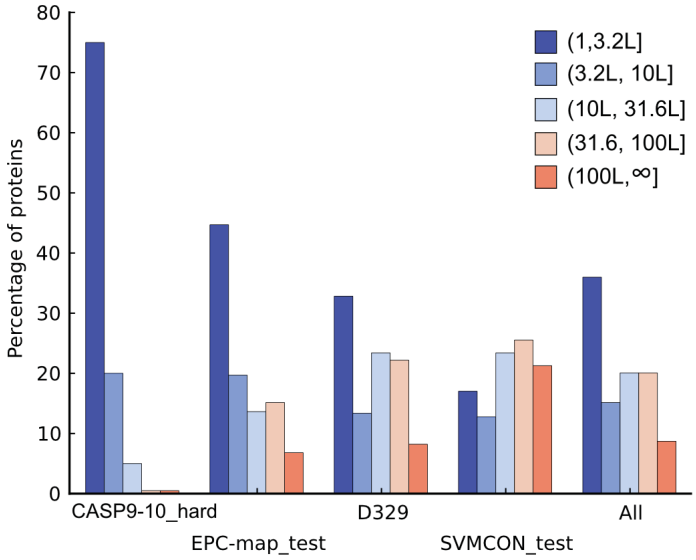

Fig. 5.5 Alignment depth composition of the CASP9-10_hard, EPC-map_test, D329 and SVMCON_test data sets. We group the proteins into bins based on their number of sequences in the alignment. Colors correspond to a particular bin, from dark blue (few sequences) to red (many sequences). We sort data sets from difficult (CASP9-10_hard) to easy (SVMCON_test). The last panel shows the pooled results. Figure source: Schneider and Brock [178].

Performance on EPC-map_test

EPC-map_test is the second difficult data set. EPC-map reaches a mean and median accuracy of 0.496 and 0.5, respectively. When compared to GREMLIN (mean: 0.363, median: 0.333), EPC-map improves the contact prediction accuracy significantly on this set (+13.3% mean accuracy; +16.7% median accuracy).

The results on this data set might represent a best-case scenario for EPC-map for several reasons. One design goal of EPC-map was robustness towards low number of sequences. Indeed, 45% of the proteins have alignments with fewer than $3.2L$ in this data set. In addition, the proteins in this data set are selected to reduce error sources that affect the *ab initio* folding of structures (see section 5.3.9). For example, internal bound ligands are a frequent error source in *ab initio* structure prediction because they are not explicitly modeled. EPC-map also might have some bias towards this data set, because its composition matches the proteins that we used for training (see section 5.3.9). Nevertheless, EPC-maps's performance on this and other data sets in this study draws a comprehensive picture of its capabilities.

Performance on D329 and SVMCON_test

The performance improvements by EPC-map are less pronounced on easier data sets (D329: +5.9%; SVMCON_test +4.8%). Since more proteins have deep alignments in this data set (Figure 5.5), sequence-based and evolutionary methods perform quite robust on this data set. However, EPC-map still improves performance by leveraging additional, physicochemical information.

Performance Summary on All Proteins

In total, EPC-map reaches 53.2% mean accuracy and 57.1% median accuracy for top $L/5$ predicted long-range contacts over 528 proteins from the CASP9-10_hard, EPC-map_test, D329 and SVMCON_test data sets. GREMLIN is the second best method with 45.4% mean accuracy and 46% median accuracy. Thus, EPC-map improves the accuracy over GREMLIN (mean accuracy: +7.8%; median accuracy +11.1%).

In addition, EPC-map more frequently predicts high-quality contact maps ($L/5$ accuracy higher than 0.3) than GREMLIN (EPC-map: 394 cases, 74%; GREMLIN: 338 cases, 64%). EPC-map also significantly improves the medium-range contact prediction accuracy (see Tables 5.4–5.7).

EPC-map achieves this superior performance by leveraging a novel source of information (physicochemical information) and integrating it with evolutionary sequence information from multiple sequence alignments.

5.4.3 Dependence of Contact Prediction Accuracy on Alignment Depth

We continue our analysis of EPC-map by investigating the performance on other factors, such as alignment depth and sequence length. We analyze these factors on all proteins from the CASP9-10_hard, EPC-map_test, D329 and SVMCON_test data sets.

In section 5.4.2, we argued that the number of sequences is a good measure for the difficulty of a protein. Here, we analyze the contact prediction performance as a function of alignment depth (Figure 5.6). Indeed, all methods benefit from more sequences. However, the impact of sequence availability is different for distinct types of contact prediction approaches. Evolutionary methods (PSICOV, GREMLIN) are strongly dependent on deep alignments. They perform poorly when less than $1L$ sequences is available, but show very high performance (0.5 mean accuracy) if more than $5L$ sequences are present. Decoy-based and machine-learning based methods are robust in the $(1, 1L]$ and $(1L, 5L]$ range, but do benefit to the same degree from sequence information as evolutionary methods. In our experiments, EPC-map improves prediction accuracy over the second best method, regardless

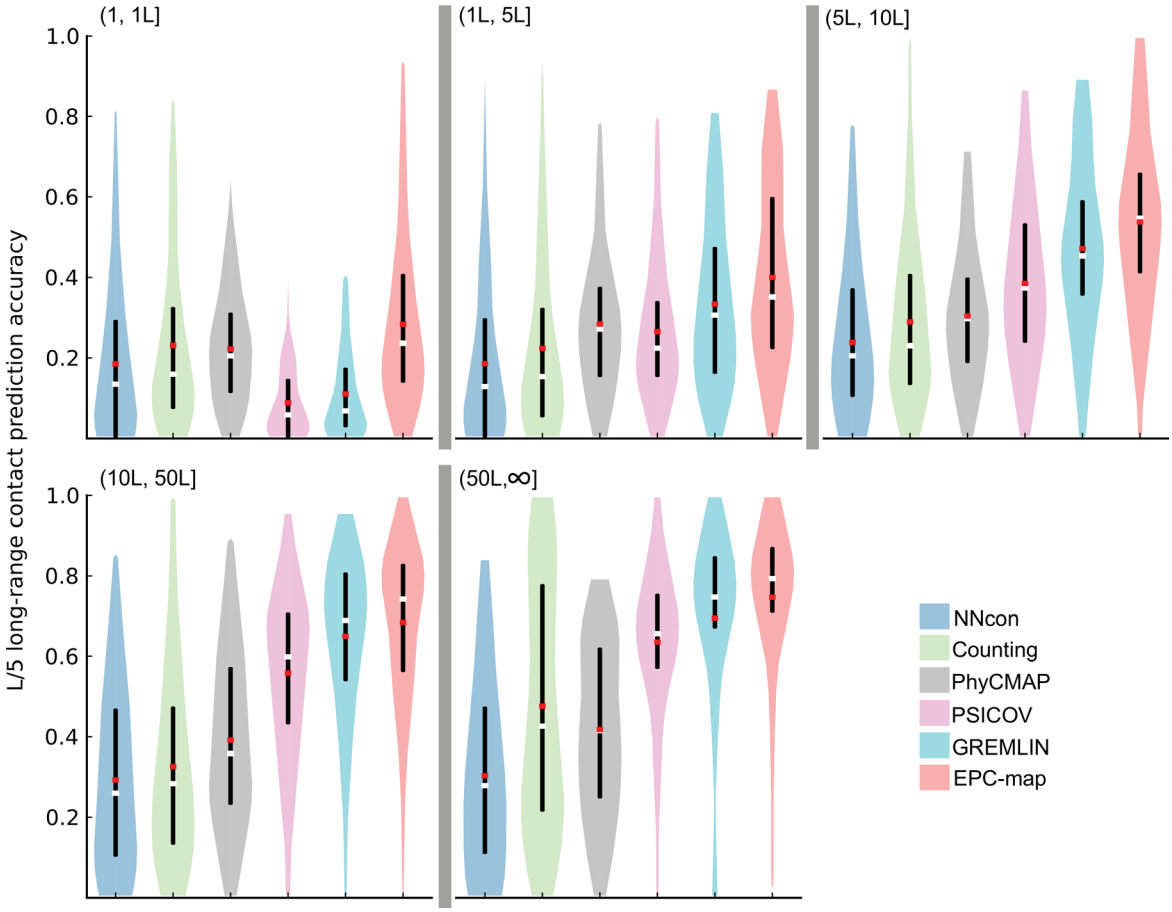

Fig. 5.6 Prediction performance for proteins with increasing sequence alignment depth. Results are shown for all proteins pooled from the CASP9-10_hard, EPC-map_test, D329 and SVMCON_test data sets. Different methods are shown as color coded violin plots. The lower and upper end of the black vertical bars in each violin denote the accuracy at the 25 and 75 percentile, respectively. White horizontal bars indicate the median, red horizontal bars the mean accuracy. The distribution of the prediction accuracies for individual proteins is indicated by the shape of the violin. EPC-map is consistently more accurate than the other tested methods, regardless how many sequences are available. Figure source: Schneider and Brock [178].

how many sequences are available. This suggests that EPC-map is a viable approach at low and high sequence availability.

5.4.4 Dependence of Contact Prediction Accuracy on Protein Chain Length

Any decoy-based method will depend on the quality of the generated decoys. The length of a protein chain is a critical factor for decoy quality and *ab initio* methods perform generally

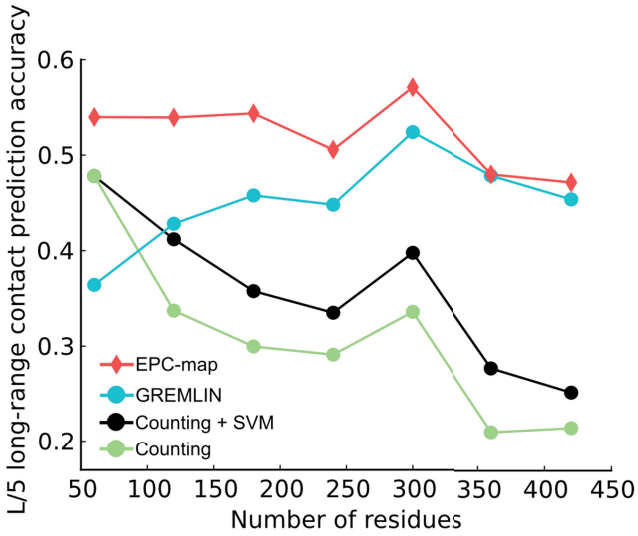

Fig. 5.7 Dependence of prediction accuracy on sequence length. EPC-map is more accurate or on par with GREMLIN, irrespective of sequence length. The performance increase over GREMLIN is most pronounced for proteins smaller than 250 residues. Counting performs better on smaller proteins. The SVM component of EPC-map consistently improves the contact prediction from decoys over Counting by leveraging physicochemical information. Figure source: Schneider and Brock [178].

well for short proteins [170]. In fact, the performance of decoy-based methods might be more susceptible to chain length than the performance of sequence-based methods. We analyze the prediction performance of EPC-map as a function of chain length in Figure 5.7. This figure also shows the performance of GREMLIN (best reference method of our analysis), Counting, and Counting+SVM (which is the SVM component of EPC-map).

EPC-map improves prediction accuracy for all proteins of varying size, but the improvement is most pronounced for proteins smaller than 250 amino acids. Counting performs better for smaller targets, because they are easier to fold and therefore result in higher quality decoys. Not surprisingly, this partially accounts for the higher performance of EPC-map for smaller proteins. In addition, Figure 5.7 shows that physicochemical information from our SVM model clearly improves decoy-based contact prediction, since the performance of Counting+SVM is consistently higher. EPC-map loses its advantage for longer proteins, probably because Rosetta fails go generate high-quality decoys for the longer chains. However, EPC-map is still slightly more accurate or as accurate as GREMLIN for larger proteins.

Table 5.8 Accuracies of the single SVM classifiers and the Ensemble SVM on 528 proteins from the CASP9-10_hard, EPC-map_test, D329 and SVMCON_test data sets. Table source: Schneider and Brock [178].

Classifier	Range	Acc[a](SE[b])/Cov[c][L/10]	Acc[a](SE[b])/Cov[c][L/5]	Acc[a](SE[b])/Cov[c][L/2]
SVM 1	Long	0.309(0.011)/0.031	0.280(0.009)/0.058	0.218(0.007)/0.113
SVM 2	Long	0.294(0.011)/0.031	0.267(0.009)/0.057	0.220(0.007)/0.114
SVM 3	Long	0.317(0.010)/0.033	0.283(0.009)/0.059	0.222(0.007)/0.115
SVM 4	Long	0.328(0.011)/0.033	0.287(0.009)/0.059	0.220(0.007)/0.135
SVM 5	Long	0.309(0.011)/0.032	0.286(0.009)/0.060	0.224(0.007)/0.117
Ensemble SVM	Long	0.387(0.012)/0.039	0.332(0.010)/0.068	0.255(0.007)/0.131

[a] Accuracy
[b] Standard error
[c] Coverage

5.4.5 Analysis of the SVM Ensemble in EPC-map

In the previous section, we showed that the SVM component of EPC-map consistently improves decoys-based contact prediction of simple occurrence-based approaches (see Figure 5.7). In this section, we analyze the performance of the SVM component in the context of the overall prediction system.

We first analyze the value of forming an SVM ensemble over the single SVMs. The contact prediction accuracy of the individual classifiers is between 0.267-0.287 (Table 5.8). The bagging approach of forming an SVM ensemble improves the prediction accuracy to 0.333. Therefore, the SVM ensemble improves prediction performance over individual SVMs, presumably because of the compensation of information loss by undersampling and reduction of variance through bagging [204]. Another benefit is that training SVMs on small sub samples is much faster than a single SVM with all training data.

Table 5.9 Contribution of the SVM component to contact prediction. Table source: Schneider and Brock [178].

Method	Range	Acc(SE)/Cov[L/10]	Acc(SE)/Cov[L/5]	Acc(SE)/Cov[L/2]
120 proteins with (1, 1L] sequences				
with SVM	Long	0.335(0.023)/0.038	0.278(0.019)/0.062	0.205(0.014)/0.110
w/o SVM	Long	0.305(0.024)/0.035	0.244(0.019)/0.055	0.188(0.015)/0.102
102 proteins with (1L, 5L] sequences				
with SVM	Long	0.471(0.025)/0.045	0.395(0.022)/0.076	0.279(0.015)/0.134
w/o SVM	Long	0.475(0.026)/0.045	0.388(0.022)/0.073	0.280(0.016)/0.133
306 proteins with >5L sequences				
with SVM	Long	0.741(0.012)/0.071	0.678(0.012)/0.131	0.530(0.011)/0.253
w/o SVM	Long	0.739(0.012)/0.070	0.679(0.012)/0.131	0.528(0.011)/0.250

In a second experiment, we also omit the SVM component from EPC-map and combine only Counting and GREMLIN scores. We re-tune the β (used in linear combination of Counting and GREMLIN) in the same way as described in section 5.3.7. The SVM component improves the mean long-range $L/5$ prediction accuracy by 3.4% if few sequences (less than $1L$) are available. Therefore, leveraging the additional information source of physicochemistry is most helpful when sequence information is absent. In these cases, sequence-based methods suffer from the unavailability of sequence information. Thus, EPC-map fills a blank spot in the spectrum of methods for contact prediction.

However, this experiment also indicates that the gain in prediction accuracy of the SVM system is less pronounced in the full EPC-map system than when combining Counting with the SVM. Currently, EPC-map implements a simple, linear combination of evolutionary information, occurrence statistics, and physicochemical information (see section 5.3.7). A more sophisticated combination of these information sources that takes properties of the protein (for example sequence length, number of available sequences, contact order) might further increase the prediction accuracy of EPC-map.

Our experiments demonstrate that the combination of physicochemical and evolutionary information advances the field of contact prediction by a simple approach that improves prediction accuracy and is robust to low sequence availability.

5.4.6 EPC-map Improves *Ab Initio* Structure Prediction

The main purpose of EPC-map is to guide *ab initio* tertiary structure prediction. Therefore, we performed a EPC-map guided structure prediction experiment to test the benefit of predicted contacts on 3D structure prediction accuracy. In this experiment, we compare the prediction performance of EPC-map *ab initio*-guided Rosetta calculations with unguided calculations.

We tested the impact of including information from EPC-map predictions into *ab initio* Rosetta calculations for the 132 proteins from EPC-map_test. To incorporated contact constraints, we add a bounded Lorentz function energy term for each contact (see section 5.3.8 for details). If a contact is satisfied in a decoy, this function assigns an energy bonus. If a restraint is significantly violated, this energy bonus falls back to zero. Therefore, contacts that are significantly violated are simply neglected. If violated constraints would be penalized, false positive contacts can have the detrimental effect of assigning high energy penalties to otherwise well folded decoys.

For each protein in EPC-map_test, we generate 1000 decoys with contact constraints and 2000 without constraints. Since 1000 decoys are already used to predict contacts using EPC-map, generating 2000 decoys without constraints allows for a fair comparison because both

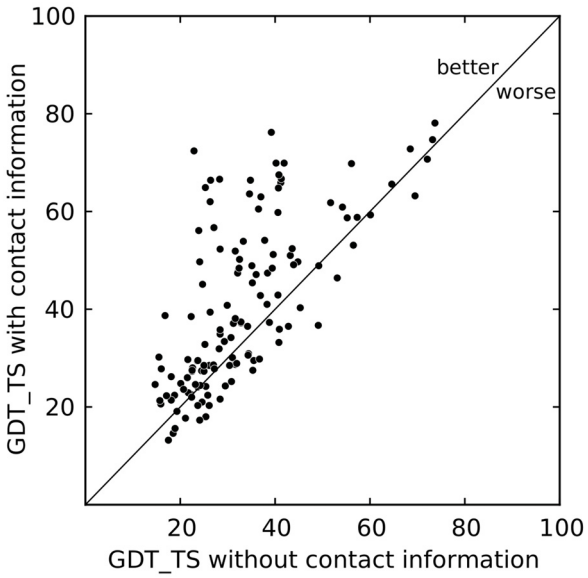

Fig. 5.8 Comparison of *ab initio* structure prediction of 132 proteins from EPC-map_test with and without predicted contacts: each data point corresponds to the GDT_TS of the lowest-energy structure generated with and without the use of EPC-map predicted contacts; EPC-map increases the GDT_TS by 7.8 from 33.1 to 40.9 GDT_TS (paired Student's t-test p-value$< 10^{-10}$). Figure source: Schneider and Brock [178].

cases have the same amount of sampling available. We also studied the accuracy/coverage trade-off of contact constraints on structure prediction accuracy. We find that $1.5L$ contacts give the best performance.

Figure 5.8 compares the structure prediction accuracy of contact-guided and unguided Rosetta calculations. We use the GDT_TS to quantify decoy quality (see Appendix A), which ranges from 0 (no similarity of compared structures) to 100 (perfect structural match). A GDT_TS of 50 or more is indicative for a decoy that captures the native topology. In our experiments, contact-guided Rosetta reaches a mean GDT_TS of 40.9, which is more accurate than standard Rosetta calculations with a mean GDT_TS of 33.1 (+7.8%). The paired Student t-test p-value$< 10^{-10}$. Therefore, the result is statistically significant. For 41/132 proteins, the GDT_TS increases by more than 10 and for 24/132 cases by more than 20. For 21/132 proteins, the GDT_TS increases from below 50 to 50 or higher. In summary, our results show that EPC-map enhances protein structure prediction performance by increasing decoy quality.

The model of one protein decreases by more than 10 GDT_TS when predicted contacts are used. In this case, EPC-map predicts mostly wrong contacts and misleads tertiary structure prediction.

5.4.7 Example Predictions with EPC-map

In this section, we discuss three example proteins for which EPC-map contacts significantly increase the prediction accuracy. All of these examples only have few sequence homologs available (less then $1.5L$). Figure 5.9 shows the contact maps of the selected proteins, the predicted structures with contact information, and the predicted structures without contact information.

Dissimilatory sulfite reductase D, PDB|1ucrA: EPC-map contacts result in the native topology of this protein (Figure 5.9A). Deviations are found in the loop regions and the most C-terminal helix is incorrectly modeled.

***E. coli* SSB-DNA polymerase III, PDB|3sxuB:** For this protein, standard Rosetta captures the general topology but does not correctly arrange the β-sheet topology (Figure 5.9B). Contacts from EPC-map allow the sampling of more native-like β-sheet topology. The C-terminus of the structure is wrongly oriented by an non-native anti-parallel β-sheet. The GDT_TS of the predicted structure increases from 33.0 to 53.9.

GIT1 paxillin-binding domain, PDB|2jx0A: The most prominent feature of this protein is a four-helix bundle (Figure 5.9C). Rosetta fails to find the fine-tuned packing and cannot model the second helix correctly (GDT_TS 36.5). EPC-map increases the GDT_TS to 58.8. The EPC-map model has errors in the loop regions and N-terminus.

Our experiments show that EPC-map has the potential to improve *ab initio* protein structure prediction.

5.5 Conclusion

In this chapter, we presented EPC-map, a method for residue-residue contact prediction that combines evolutionary information with physicochemical information from *ab initio* structure prediction decoys. The combination of these two sources of information improves contact prediction accuracy when compared to state-of-the-art algorithms.

The key of our approach is a graph-based encoding of the local contact network in protein decoys, which leverages physicochemical information. We distill critical features of the graphs into a vector-based representation. This forms the input for an SVM model that distinguishes native from non-native contacts in *ab initio* decoys.

Our results demonstrate that the combination of multiple information sources boosts contact prediction accuracy. We demonstrate this on extensive experiments on 528 proteins

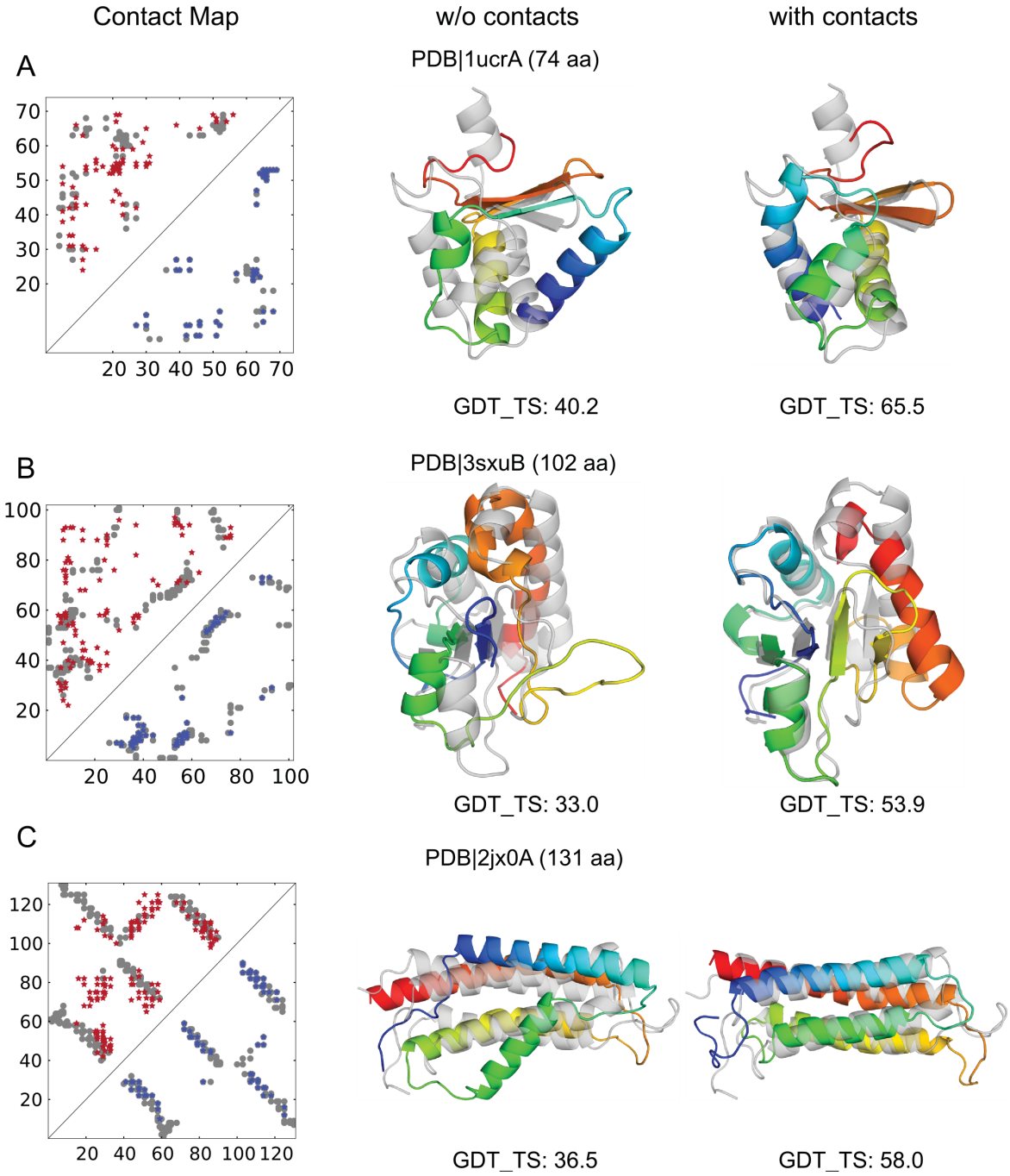

Fig. 5.9 Tertiary structure prediction improvement of the dissimilatory sulfite reductase D (PDB|1ucrA, panel **A**), of the *E. coli* SSB-DNA polymerase III (PDB|3sxuB, panel **B**), and of the GIT1 paxillin-binding domain (PDB|2jx0A, panel **C**). Contact maps show false positive predictions in the upper triangle (red), true positive predictions in the lower triangle (blue) and native contacts in grey. For the shown predictions, native structures are shown in grey and predicted structures are colored from N-terminus (blue) to C-terminus (red). The predictions correspond to the lowest-energy structure generated without use of contacts (middle column) and with EPC-map predicted contacts (right column). Figure source: Schneider and Brock [178].

and show that EPC-map achieves 53.2% mean accuracy for top $L/5$ predicted long-range contacts, 7.8% higher than the second-best method (see section 5.4.2 on page 72).

In addition, we show that the combination of two information sources compensates the performance degradation of a single, possibly poor quality, information source. Thus, leveraging physicochemical information alleviates the main short-coming of evolution-based prediction methods: The high dependence on homologous sequences. Our results show that EPC-map is effective even when deep alignments are not available (see Figure 5.6 on page 73 and Table 5.9 on page 75).

Our structure prediction experiments indicate that EPC-map contacts improve the GDT_TS of *ab initio* models. Importantly, one can often find only few sequences for many protein targets for which *ab initio* structure prediction is necessary (no templates). Therefore, combination methods to residue-residue contact prediction are a promising route towards solving the general structure prediction problem.

Encouraged by our results, we incorporated EPC-map into our novel structure prediction pipeline, RBO Aleph. We tested the contact and *ab initio* structure prediction capabilities of RBO Aleph in a blind manner in the CASP11 experiment. The next chapter describes the RBO Aleph pipeline and our CASP11 results.

Chapter 6

Analysis of Contact and Free Modeling Predictions by RBO Aleph in CASP11

The RBO Aleph server is originally described in:

Mabrouk, M.*, Putz, I.*, Werner, T., Schneider, M. Neeb, M., Bartels P. and Brock, O. (2015). RBO Aleph: leveraging novel information sources for protein structure prediction. Nucleic Acids Res., 43(W1):W343–W348.

*contributed equally

Own contributions: My particular contribution was the design and implementation of EPC-map and contact-guided model-based search (main developer from 2010-2015) into the *ab initio* pipeline of RBO Aleph. I also conceived and designed the server. I designed and implemented significant parts of the server, especially on the backend. I maintained the server during CASP11 and contributed to paper writing.

Contributions of co-authors: MM, IP, TW, and OB conceived and designed the server. MM, IP, TW designed and implemented significant parts of the server. MM, IP, TW, PB, and MN designed and implemented the frontend of the web server. MM, IP, and TW maintained the server during CASP11. MM, IP, TW, and OB contributed to paper writing.

This chapter is based on the following publication:

Mabrouk, M., Werner, T., Schneider, M. Putz, I., and Brock, O. (2015) Analysis of Free Modeling Predictions by RBO Aleph in CASP11. Proteins, in press.

Own contributions: I conceived and designed experiments. I performed experiments and developed analysis tools. I analyzed the data, with a focus on the residue-residue contact prediction results of RBO Aleph in CASP11 and the effect of component combinations

on the pipeline output. The contact prediction results of RBO Aleph are based on the algorithm developed in chapter 5. I was the main developer of the conformational space search algorithm of RBO Aleph, contact-guided model-based search, from 2010-2015. I contributed to paper writing.

Contributions of co-authors: MM, TW, IP, and OB conceived and designed experiments. MM, TW, and IP performed experiments. MM, TW, and IP conceived and implemented analysis tools. MM, TW, and IP analyzed data. MM, TW, IP, and OB contributed to paper writing. The following figures and tables were prepared in part or in modified form by co-authors of the original paper: MM: Figure 6.6, 6.9. TW: Figure 6.7, 6.8, 6.11, 6.12, Table 6.4. IP: Table 6.1.

Extensions: This chapter extends the original paper by implementation details of model-based search and contact-guided model-based search (guided-MBS).

6.1 Introduction

In this section, we present the results of blind-testing EPC-map (see chapter 5) in the 11th community-wide Critical Assessment for Protein Structure Prediction experiment (CASP11). We integrated EPC-map into an automated structure prediction server, RBO Aleph, a complete pipeline to protein structure prediction (Figure 6.1). My main contribution to RBO Aleph was the integration of two unique features: Residue-residue contact prediction by EPC-map and conformational space search by contact–guided model-based search (guided-MBS). Thus, this dissertation thoroughly describes the *ab initio* pipeline of RBO Aleph and focuses on the analysis of our contact and *ab initio* structure prediction results in CASP11.

In CASP11, *ab initio* structure prediction is mainly applied in the free-modeling category. Free-modeling targets are proteins for which homologous structures (templates) do not exist or are difficult to find by fold recognition[1]. In this difficult free-modeling category, RBO Aleph ranked as one of the best automated servers: RBO Aleph ranked first by average Z-score > 0 and third by sum Z-score > -2 (ranking based on first submitted models and assessors' formula). We set out to analyze to what degree the unique components of RBO Aleph, EPC-map and contact–guided MBS, contributed to our CASP11 performance.

First, we analyze the performance of EPC-map as a contact prediction method. EPC-map reached state-of-the-art contact prediction performance: Rank two for medium+long-

[1]Note that this category is defined by the target class and not by the nature of the algorithm that predictors use to model these proteins. Therefore, methods that compete in this category might use *ab initio* modeling, homology modeling, or a mix of both.

range contacts (sequence separation > 11) and rank five for long-range contacts (sequence separation > 23).

To isolate the contribution of conformational space search by MBS, we repeated the MBS calculations *without* contacts analyzed the resulting decoy ensembles. MBS finds lower energy decoys than Rosetta's Monte-Carlo-based conformational space search algorithm [170], which leads to higher GDT_TS predictions in nine out of 18 cases.

The most encouraging result, however, is that EPC-map contacts increase the free-modeling performance of RBO Aleph pipeline by 28.8%[2]. This demonstrates the high utility of EPC-map contacts in *ab initio* structure prediction, especially when used in synergy with guided-MBS.

Finally, we analyze the impact of the *ab initio* components—EPC-map contacts, model-based search, and decoy selection—on the final prediction outcome of the RBO Aleph pipeline (GDT_TS of resulting first and best of five models). Our analysis reveals a possible fundamental problem in evaluating the performance of protein structure prediction methods: Improvements of individual components do not necessarily translate into improvements of the entire pipeline. This suggests that the interactions between the components are poorly understood, which is a significant impediment to community-wide progress.

Section 6.2 introduces the RBO Aleph pipeline with an emphasis on the components that are relevant to free-modeling. Implementation details follow in section 6.3. In section 6.4, we present a detailed analysis of RBO Aleph in CASP11, including contact prediction results of EPC-map, analysis of MBS and guided-MBS, and a detailed pipeline-level analysis. Section 6.5 concludes this chapter.

6.2 Overview of RBO Aleph

In this section, we give an overview of the RBO Aleph pipeline and explain the interplay of various protein structure prediction methods. However, the main contribution of this dissertation is the modeling of difficult *ab initio* targets that are lacking templates. Therefore, we explain the algorithm for *ab initio* structure prediction in more detail.

RBO Aleph implements a complete protein structure prediction pipeline to model a wide range of protein targets. This includes multi-domain targets, template-based targets, and *ab initio* targets (see Figure 6.1). RBO Aleph first predicts contacts with EPC-map. It then retrieves templates with various template retrieval programs. Then, the server evaluates template quality and decides to model the structure with template-based or *ab initio* structure prediction. Finally, RBO Aleph ranks the models with a knowledge-based potential and sub-

[2]Measured by sum Z-score of the first model.

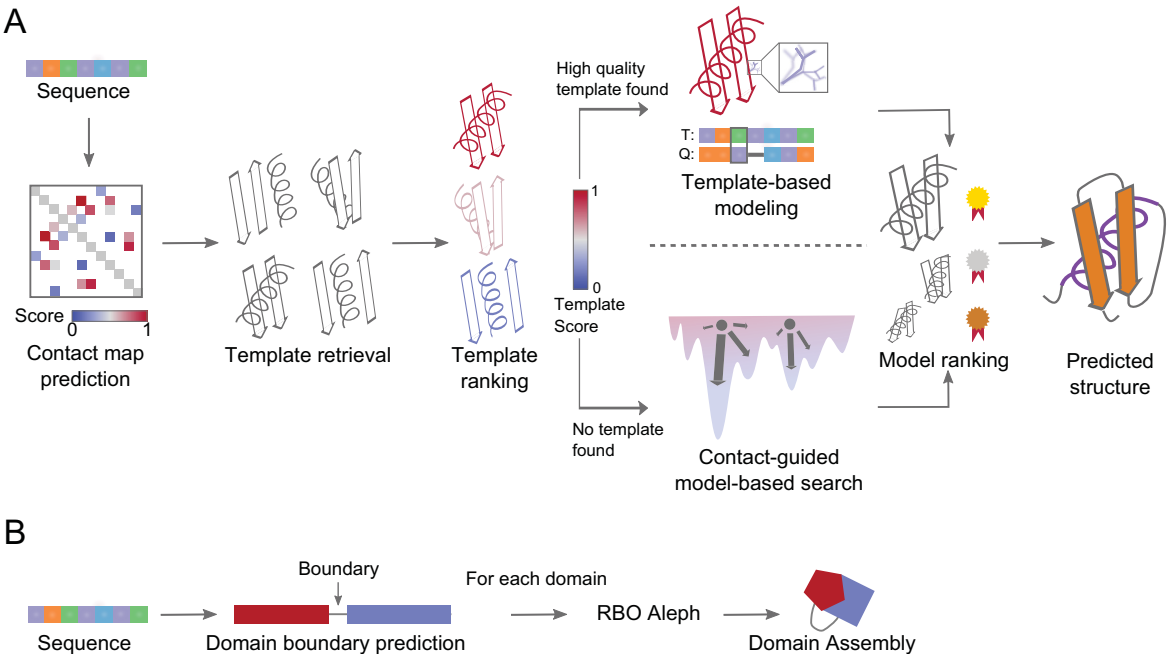

Fig. 6.1 Overview of the RBO Aleph pipeline. **A**: Overview of the RBO Aleph pipeline (excluding domain boundary prediction for the sake of clarity). The server takes a protein sequence as input. First, EPC-map predicts residue–residue contacts [178]. Then, RBO Aleph identifies template structures with several threading algorithms (see methods). A random forest classifier re-ranks the templates. If suitable templates are identified, RBO Aleph performs template-based modeling with MODELLER [4]. If no templates are available, RBO Aleph performs *ab initio* structure prediction with contact-guided model-based search. The server than ranks the models with ProSA [193] and submits the top five structures as the final predictions. **B**: Overview of the RBO Aleph domain boundary and assembly pipeline. RBO Aleph uses the consensus of PPRODO [188] and DomPro [39] to predict domain boundaries. Figure adapted from: Mabrouk et al. [125].

mits the final prediction (see Figure 6.1A). RBO Aleph also implements a domain boundary prediction step to model multi-domain proteins (see Figure 6.1B). If the pipeline detects a multi-domain protein, it models each domains independently with the full RBO Aleph pipeline, and assembles the domain models in a last step.

In the next section, we provide an overview of the *ab initio* modeling pipeline of RBO Aleph.

6.2.1 *Ab Initio* Modeling Pipeline

In the *ab initio* modeling pipeline of RBO Aleph, we first predict residue–residue contacts with EPC-map. We then use these contacts to guide model-based conformational space search (MBS) [26]. For the specific purpose of guiding *ab initio* structure prediction with

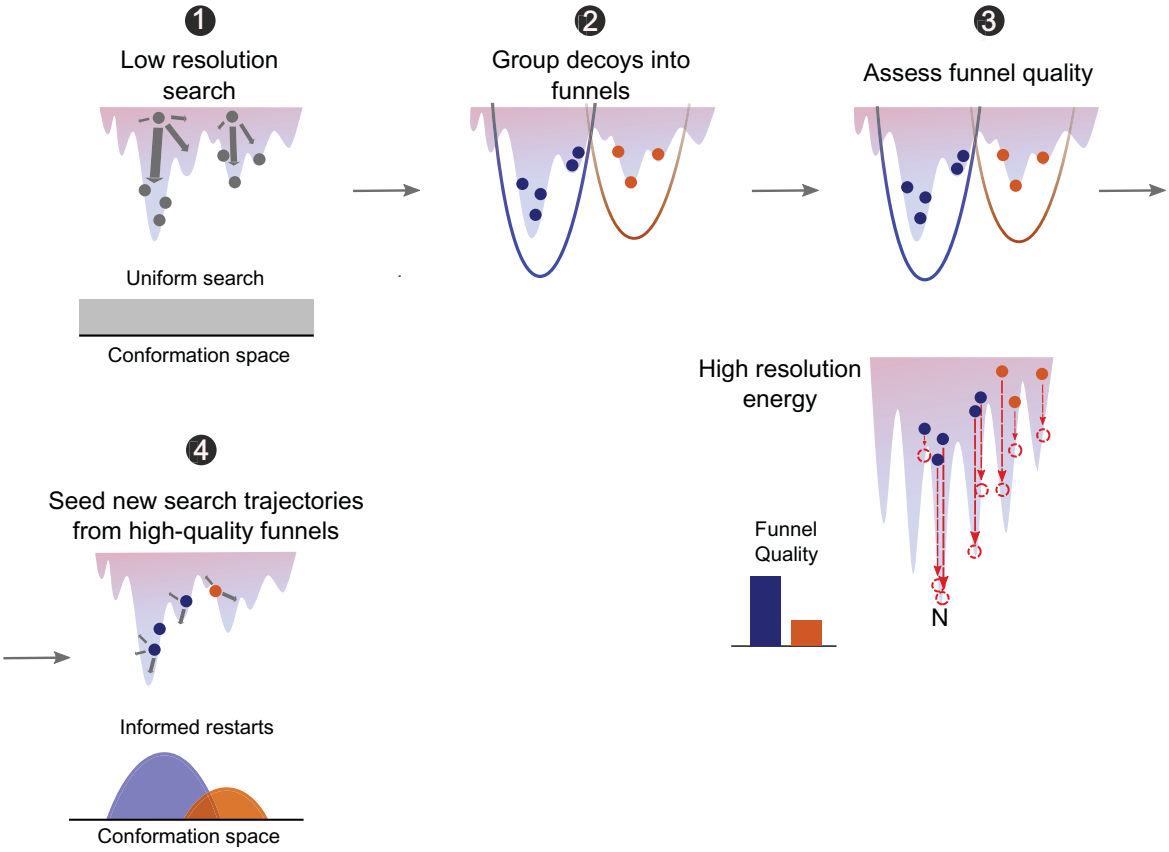

Fig. 6.2 Outline of the model-based search algorithm (MBS). This figure illustrates a single iteration of MBS. **1**: Initially, MBS starts to fold structures from the extended chain. In the first MBS stage, this corresponds to random, uniform Monte Carlo search over the entire search space. **2**: MBS groups decoys into funnels with a heuristic clustering algorithm. **3**: For the five lowest-energy decoys in each funnel, MBS switches to a high-resolution phase and refines the decoys in an accurate all-atom energy. The quality of a funnel is proportional to the lowest all-atom energy found in that funnel. **4**: For the next iteration, MBS eliminates the worst 50% funnels. This stops the search trajectories of the decoys in these funnels. The search trajectories are then restarted from the decoys in the remaining funnels. MBS allocates computational resources proportional to the funnel quality. This means that MBS launches more search trajectories from high quality funnels.

contact information, we devised a modified contact-guided model-based search algorithm. In the final step, RBO Aleph ranks the resulting decoys using a knowledge-based potential.

We already described the EPC-map algorithm in the previous chapter. Thus, we start the method overview with an introduction to the model-based search algorithm and its contact-guided variant.

6.2.2 Model-Based Search

Model-based search is a heuristic search algorithm that identifies promising regions in conformational space and allocates computational resources to increase sampling in these regions. Here, we provide a high-level overview of the algorithmic principle of MBS. A full description of the implementation can be found in the original paper [26]. Figure 6.2 details a single iteration of MBS. MBS launches short trajectories that explore the conformational space with Monte Carlo sampling. These search trajectories sample protein structures in a reduced representation (side-chains are modeled as centroid pseudo-atom) using a low-resolution energy function. MBS clusters the resulting structures into funnels with a heuristic clustering algorithm. MBS then adds side-chains to the five lowest-energy structures in each funnel and refines them in an all-atom energy landscape. MBS then judges the quality of each funnel by the all-atom energy of the lowest energy structure that is part of that funnel. Importantly, the all-atom energy is more accurate than the low-resolution energy. Thus, judging the quality of a funnel by all-atom energy uses higher quality information. MBS uses this high quality information to restart trajectories from seed structures of the best funnels.

6.2.3 Contact-Guided Model-Based Search

RBO Aleph integrates a modified model-based search algorithm that is adapted for search with noisy constraints (guided-MBS, see Figure 6.3). Guided-MBS represents constraints from EPC-map by a modified Lorentzian function (see section 5.3.8).

Guided-MBS adds contact information to the energy function in the low-resolution phase (Figure 6.3A). The contacts steer search towards biologically relevant regions in conformational space and deepen wells where they are approximately satisfied. Guided-MBS also uses contact constraints to sort the decoy structures before clustering them into funnels. Here, the lowest energy structure is used as a funnel seed (please refer to Brunette and Brock [26] for details on the heuristic clustering algorithm of MBS).

Although contacts guide search in the low-resolution phase, they might distort the all-atom energy landscape by reducing its resolution [166]. Furthermore, if contacts are noisy due to low prediction accuracy, the constrained energy landscape will contain errors and

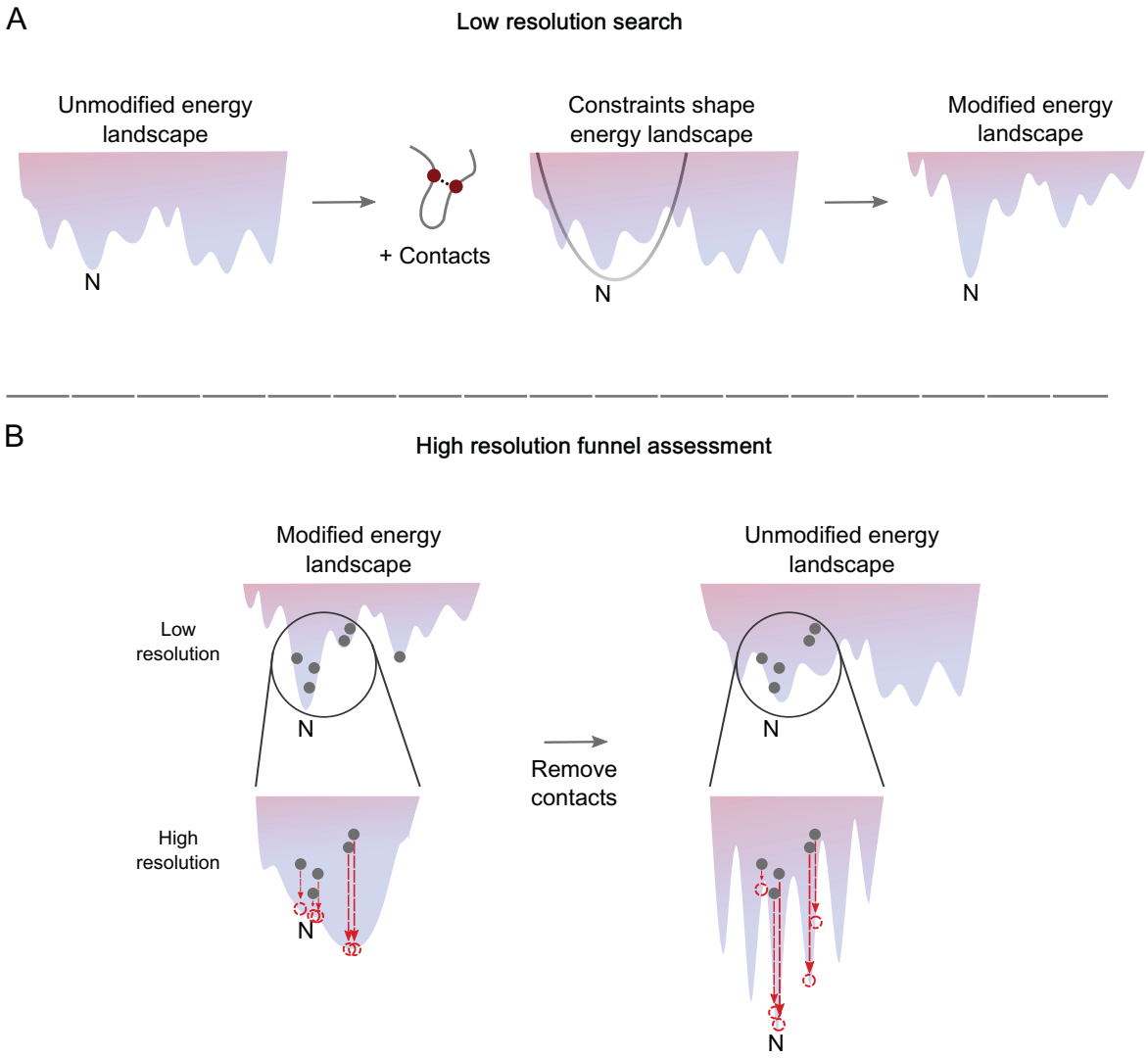

Fig. 6.3 Illustration of contact-guided model-based search (guided-MBS). **A**: Guided-MBS adds contact constraints to the energy landscape, effectively reshaping the low-resolution energy landscape by deepening wells where the constraints are satisfied. **B**: In the high-resolution phase, the noise in predicted contacts, which usually contain false positives, might distort the energy landscape. This results in reduced resolution and eventually disassociates low-energy regions with the location of the native structure. To alleviate this issue, MBS removes the contacts prior to the high-resolution phase. Therefore, guided-MBS refines and scores decoys by the unbiased all-atom energy.

steer search into the wrong direction. To bypass this issue, we remove contact constraints in the high-resolution phase and minimize the structures in the unbiased Rosetta all-atom energy [170] (Figure 6.3B). Note that we also assess funnel quality by the unbiased all-atom energy. By decoupling the contacts from all-atom refinement and funnel assessment, we increase the influence of the information encoded in the all-atom energy function. This might partially compensate for errors that are made when searching the conformational space with highly noisy contacts.

In this section, we presented an overview of the conformational space search technique in RBO Aleph, guided-MBS. In addition to EPC-map, this method forms the unique features of the RBO Aleph pipeline. In the next section, we provide implementation details of all RBO Aleph components.

6.3 Implementation of RBO Aleph

6.3.1 Contact Prediction

RBO Aleph predicts residue-residue contacts with EPC-map, as described in chapter 5 (sections 5.2 and 5.3 on page 50 and 54).

6.3.2 Template-Based Modeling Pipeline

For template-based modeling, RBO Aleph first searches templates with the following threading programs: LOMETS [225], SPARKS-X [230], RaptorX [158], and HHPred [78]. In the next step, RBO Aleph re-ranks the templates with a random-forest classifier [24]. The random forest classifier is trained on the output scores from the threading programs, on structure similarity between template structures, and the match of EPC-map contacts to the templates. RBO Aleph considers all templates above a probability threshold of 0.4 of the random-forest classifier. After selecting the template with the highest score, the next template is added to the template pool if it has at least a TM-score of 0.7 to the next best template and a random forest score of 0.8 times the score of the best template. If the next scoring template is not similar to the higher scoring template (TM-score < 0.7), RBO Aleph only accepts the template if it does not overlap with all previous accepted templates. This ensures that templates are not too diverge if they model the same part of the protein, because structural heterogeneity reduces the effectiveness of the homology protocol. Furthermore, this increases the template coverage of the protein.

If RBO Aleph detects templates, it performs homology modeling using MODELLER [4]. The server models all parts of the sequence that are not covered by templates with a Rosetta loop modeling protocol [82].

6.3.3 Implementation of Guided-MBS

We integrate MBS into the Rosetta release version 3.4. This allows us to employ Rosetta's routines for fragment assembly and its energy functions [170]. Rosetta performs Monte Carlo sampling and uses structural fragments to move in structure space. In a preprocessing step, Rosetta selects these fragments for the target protein using sequence similarity, sequence profiles, and secondary structure information [73]. We use fragments of length three and nine in MBS.

We implement the MBS protocol in six stages. Each MBS stage generates multiple decoys using a message passing interface (MPI) implementation. On the processing nodes, MBS uses the Monte Carlo sampling protocol from a particular stage in the Rosetta `AbInitioRelax` protocol. The Monte Carlo protocols differ in the fragment length and in low-resolution energy term weights. MBS stage 1 starts from the extended chain and builds a coarse, initial topology. MBS stage 1 uses stage 1 and 2 of the Rosetta `AbInitioRelax` protocol. The next four MBS (2-5) stages sample the conformational space with 9-mer fragment insertions of Rosetta's stage 3. The last MBS stage (6) uses Rosetta stage 4 to refine structures with 3-mer fragment replacement. The exact weightings and implementation details of the Rosetta stages are documented elsewhere [170].

After the sampling step of each stage, MBS performs heuristic clustering to construct funnels. It then refines the five lowest-energy decoys in each funnel with a realistic, hybrid all-atom physical/knowledge-based force field. Implementation details of the clustering procedure are given in the original MBS paper [26].

The sampling and clustering steps use contact constraints from EPC-map with the Lorentz function. We use the same parameters as in section 5.3.8, albeit with $1.5L$ contacts (L is the length of the sequence), which we empirically found to maximize the GDT_TS of the first model.

6.3.4 Decoy Selection in RBO Aleph

The last step of the RBO Aleph pipeline is the selection of models for submission. RBO Aleph employs a knowledge-based potential (ProSA) [193] to select models. In addition, RBO Aleph estimates the local error of the models with ModFOLD [133]. RBO Aleph submits the five highest ranked predictions (as estimated by ProSA).

6.3.5 Domain Boundary Prediction and Assembly

Multi-domain proteins consist out of two or more domains that might vary in their template availability. Therefore, a different protocol (*ab initio* or template-based) might be appropriate for the modeling of an individual domain. RBO Aleph detects domain boundaries and predicts the structure of a domain with an independent RBO Aleph run that again estimates the template availability for the predicted domain and selects the appropriate protocol. RBO Aleph employs PPRODO [188] and DomPro [39] for domain boundary prediction. RBO Aleph predicts domain boundaries by the consensus score of these two sequence-based domain boundary predictors. Since we process each domain with an individual RBO Aleph run, the domain sequence might be recursively split until no further domains are detected. After modeling of the individual domains, RBO Aleph assembles the models of each domain into the full-length structure with the domain assembly protocol described by Wollacott et al. [221]. In the last step, ProSA ranks the full-length models as explained in the previous section (section 6.3.4).

6.3.6 Post-CASP11 Analysis of RBO Aleph

We performed several post-CASP11 experiments to analyze the behavior of the RBO Aleph pipeline. This analysis is limited to proteins for which we had native structures available at the time of this study. This was the case for 27 domains. RBO Aleph modeled 20 out of 27 domains in CASP11 using the *ab initio* modeling pipeline. Note that we did post-CASP predictions experiments of the *ab initio* part of the pipeline only on the 20 proteins that we processed with *ab initio* structure prediction. This mimics a modified *ab initio* pipeline that does not influence the decision whether RBO Aleph processes a domain with the *ab initio* protocol. The target T0831 was excluded due to technical reasons. Overall, this results in 17 target proteins with 18 FM domains defined by RBO Aleph and 20 FM domains by the official CASP11 domain definition. Table 6.1 shows detailed information about these targets and domain assignments.

We use the 18 RBO Aleph domains for the analysis of MBS as a search strategy. For some targets, this domain definition differs from the official domains. However, this mimics the real prediction scenario, in which we have no access to the "true" domain definitions. We use official CASP11 domain assignments to evaluate the performance of EPC-map and the entire RBO Aleph pipeline. This mimics the assessment of the RBO Aleph pipeline under CASP conditions.

Table 6.1 Analyzed CASP11 targets with at least one FM-domain. For each target we list the total number of detected domains together with the FM-domain definitions assigned by RBO Aleph. The last column contains the PDB-ID of the native structure if available. Table source: Mabrouk et al. [126].

	Target	$\#D_{RBO}$	FM-Domain(s)$_{RBO}$	$\#D_{CASP}$	FM-Domain(s)$_{CASP}$	PDB
1	T0761	1	D1: 1-285:285	2	D1: 62-149:88	4pw1
				2	D2: 150-285:136	
2	T0763	1	D1: 1-163:163	1	D1: 31-160:130	4g0y
3	T0767	1	D1: 1-318:318	2	D1: 133-312:180	4qpv
4	T0771	1	D1: 1-204:204	1	D1: 27-76,91-191:151	4qe0
5	T0777	1	D1: 1-366:366	1	D1: 118-362:345	-
6	T0781	1	D1: 1-421:421	1	D1: 40-240:200	4qan
7	T0785	1	D1: 1-115:115	1	D1: 3-114:112	4d0v
8	T0791	1	D1: 1-300:300	2	D1: 6-161:156	4kxr
					D2: 162-300:139	
9	T0794	2	D2: 373-470:98	2	D2: 291-462:172	4cyf
10	T0806	1	D1: 1-258:258	1	D1: 1-256:256	-
11	T0808	2	D2: 140-418:278	2	D2: 150-418:269	4qhw
12	T0824	1	D1: 1-110:110	1	D1: 2-109:108	-
13	T0832	2	D1: 1-51:51	1	D1: 10-218:209	4rd8
			D2: 52-257:206			
14	T0834	1	D1: 1-219:219	2	D1: 2-37,130-192:99	4r7q
					D1: 38-129:92	
15	T0836	1	D1: 1-204:204	1	D1: 1-204:204	-
16	T0837	1	D1: 1-128:128	1	D1: 1-121	-
17	T0855	1	D1: 1-119:119	1	D1: 5-115	2mqd

6.4 Results and Discussion

We now present our analysis of RBO Aleph in CASP11. Since the major contributions of this thesis to RBO Aleph are the development of EPC-map and guided-MBS, our analysis focuses on these components.

First, we analyze the performance of EPC-map in the CASP11 blind test. We then analyze the free-modeling performance of the RBO Aleph server, with an emphasis on the performance of the MBS search strategy. Additionally, we investigate the impact of EPC-map contacts on MBS (guided-MBS). We then highlight three above and below-average predictions of RBO Aleph (measured by Z-score). These predictions represent particular noteworthy successes and challenges of our *ab initio* protocol. Finally, we analyze the interaction between the components of the RBO Aleph pipeline by systematically comparing different pipeline configurations.

6.4.1 Performance of EPC-map in CASP11

RBO Aleph submits EPC-map contacts directly for the residue-residue contact prediction category. In contrast to the results in chapter 5, the results presented in this chapter are from a blind test of EPC-map. In this blind test, the structures of the proteins were unknown to us at the time of prediction. The advantage of blind testing is that it removes some of the bias that researchers might (unconsciously) build into the experimental setups in which they test their methods.

Figures 6.4 and 6.5 show the performance of EPC-map in CASP11. EPC-map predicted contacts for 39 evaluation domains[3]. The analysis in this section is based on official data from CASP11 and on the official CASP11 domain assignments (data taken from http://www.predictioncenter.org). Unless noted otherwise, the results presented here refer to the top $L/5$ predicted contacts on free-modeling domains.

First, we report the performance and ranking of EPC-map. We quantify the performance by the per-target Z-score (based on the $L/5$ contact accuracy). A positive Z-score indicates that EPC-map performed better than the average CASP11 predictor (Figure 6.4A). Tables 6.2 and 6.3 summarize accuracy, Xd values, and Z-scores for each target domain. EPC-map predicted more accurate long-range contacts than the average CASP11 predictor (positive Z-scores) for 17/36 domains (47.2%). For medium+long-range contacts, EPC-map reached positive Z-scores in 34/39 (87.1%) cases. Overall, EPC-map ranks second for medium+long-range contacts and fifth for long-range contacts in CASP11 (Figure 6.4B).

We now directly compare the leading method of CASP11 (CONSIP2 [105]) with EPC-map. We also compare the prediction accuracy of EPC-map with the mean accuracy of the top five contact prediction groups (excluding EPC-map, Figure 6.5). We compare only targets for which EPC-map and CONSIP2 (or the other top five groups) submitted predictions. This accounts for differences in performance that are caused by missed targets. Note that this results in slightly different numbers than the CASP11 website reports. EPC-map reaches an average medium+long contact prediction accuracy of 43.2% for 39 evaluation domains. CONSIP2 reaches a comparable prediction accuracy of 43.0% for the same 39 evaluation domains. The top five methods reach an average medium+long contact prediction accuracy of 37.3%. The mean accuracy for long-range contacts is 13.7/25.5/18.8% for EPC-map/CONSIP2/mean top 5 groups, respectively.

Overall, EPC-map performs comparably to the top group in CASP11 for medium+long range contacts. For long-range contacts, EPC-map is among the top five methods. Since CASP11 is a true blind test, this is important evidence that physicochemical information

[3]To be precise, we submitted contacts for 72 domains, but only 39 of them lack templates and are therefore evaluated in the contact prediction category.

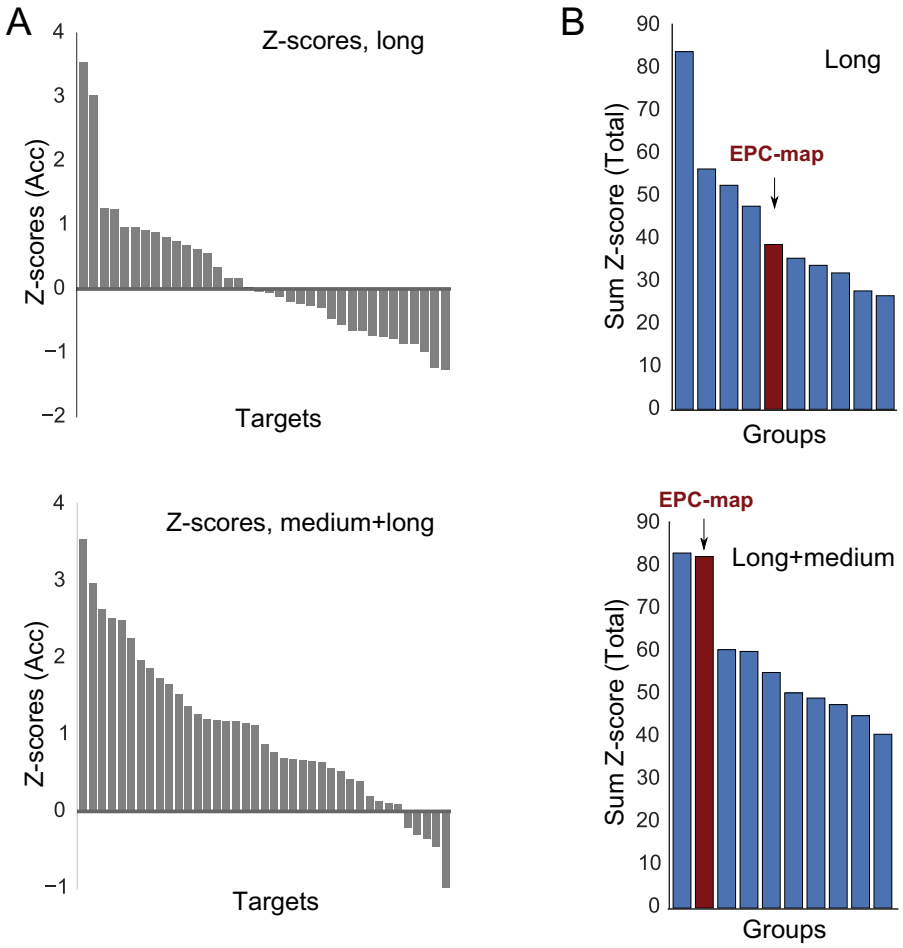

Fig. 6.4 Performance of EPC-map in CASP11. All results in this Figure refer to the top L/5 predicted contacts. All targets in this Figure are free-modeling targets. Results shown for long (sequence separation > 23) range and long+medium (sequence separation > 11) range contact prediction accuracy of EPC-map. **A**: Z-scores for individual targets of RBO-Aleph. **B**: Ranking of the top ten contact prediction methods by the sum of Z-scores (total, i.e. Acc+Xd). RBO-Aleph (EPC-map) ranks second for medium+long range contacts and fifth for long-range contacts. Figure adapted from: Mabrouk et al. [126].

Table 6.2 Detailed long-range (sequence separation $>$ 23 residues) contact prediction results of EPC-map for free-modeling target domains in CASP11. Table source: Mabrouk et al. [126].

Target	Domain	L/5 Accuracy	Z-score (Acc)	L/5 Xd	Z-score (Xd)
T0837	D1	91.67	3.54	30.44	2.92
T0806	D1	62.75	1.26	25.37	1.14
T0767	D2	38.89	0.55	19.35	0.59
T0834	D2	29.41	3.02	14.89	2.73
T0808	D2	22.22	0.91	14.11	1.1
T0791	D1	20.0	-0.23	8.47	-0.25
T0836	D1	19.51	0.61	20.0	1.43
T0785	D1	18.18	0.81	10.67	0.64
T0855	D1	17.39	1.25	12.74	1.53
T0775	D4	16.67	0.33	12.48	1.01
T0810	D1	13.04	0.75	9.21	0.94
T0802	D1	13.04	-0.03	10.3	0.94
T0761	D2	13.04	0.88	8.67	0.41
T0777	D1	11.59	0.96	10.55	0.71
T0820	D1	11.11	0.96	9.1	1.03
T0814	D1	11.11	-0.65	6.33	-0.1
T0831	D2	10.26	0.68	5.77	0.33
T0827	D2	10.0	0.17	9.97	0.52
T0814	D2	8.7	-0.98	6.6	-0.85
T0763	D1	7.69	-0.2	4.88	0.17
T0791	D2	7.14	-0.46	8.88	-0.11
T0771	D1	6.67	-0.06	8.08	0.68
T0794	D2	5.88	-1.23	1.1	-1.37
T0793	D1	5.0	-0.12	2.44	-0.33
T0826	D1	5.0	0.02	3.36	0.01
T0832	D1	4.76	0.16	6.21	0.61
T0824	D1	4.55	-0.74	3.89	-0.41
T0799	D1	3.57	-0.29	6.51	-0.11
T0804	D2	3.33	-0.78	4.67	-0.34
T0781	D1	2.5	-0.26	7.27	1.09
T0775	D2	0.0	-0.72	2.51	-0.92
T0775	D5	0.0	-0.55	7.03	0.51
T0761	D1	0.0	-0.85	3.84	0.27
T0793	D2	0.0	-0.85	-3.87	-1.28
T0834	D1	0.0	-0.65	1.96	0.46
T0793	D5	0.0	-1.26	-0.66	-1.69

Table 6.3 Detailed long+medium-range (sequence separation > 11 residues) contact prediction results of EPC-map for free-modeling target domains in CASP11. Table source: Mabrouk et al. [126].

Target	Domain	L/5 Accuracy	Z-score (Acc)	L/5 Xd	Z-score (Xd)
T0837	D1	91.67	3.54	30.25	2.68
T0855	D1	91.3	1.73	27.39	1.25
T0775	D2	76.92	0.88	24.29	0.76
T0806	D1	74.51	1.53	28.05	1.33
T0761	D1	72.22	2.97	23.0	2.22
T0761	D2	69.57	0.77	24.49	0.7
T0791	D2	67.86	1.19	26.49	1.12
T0775	D4	66.67	0.68	21.57	0.79
T0808	D2	66.67	1.37	26.12	1.17
T0791	D1	66.67	1.15	27.39	1.13
T0802	D1	65.22	2.64	22.33	2.09
T0804	D1	57.14	2.49	20.04	1.99
T0771	D1	56.67	1.97	18.13	1.09
T0814	D2	56.52	0.2	18.95	0.12
T0781	D1	55.0	2.26	22.49	1.97
T0793	D1	50.0	0.09	16.11	0.03
T0785	D1	45.45	2.52	18.45	2.17
T0834	D1	45.0	1.12	20.17	1.37
T0814	D1	44.44	0.57	16.43	0.62
T0767	D2	44.44	0.7	19.85	0.59
T0827	D2	43.33	1.66	20.77	1.49
T0793	D5	42.86	1.18	17.54	1.14
T0794	D2	41.18	0.67	18.19	0.69
T0834	D2	41.18	1.86	19.51	2.07
T0763	D1	38.46	0.39	11.56	0.1
T0775	D5	31.03	0.65	12.64	0.37
T0804	D2	30.0	1.2	14.24	1.28
T0836	D1	21.95	0.52	20.67	1.14
T0799	D1	21.43	0.11	10.82	-0.1
T0831	D2	20.51	0.64	13.93	0.51
T0832	D1	16.67	1.18	14.63	1.5
T0820	D1	16.67	1.27	11.19	1.31
T0775	D6	14.29	-0.36	8.25	-0.29
T0777	D1	13.04	0.42	14.64	0.99
T0793	D2	11.11	-0.99	2.69	-0.89
T0824	D1	9.09	-0.46	5.78	-0.18
T0826	D1	5.0	0.13	4.09	-0.06
T0810	D1	4.35	-0.22	5.8	0.46
T0799	D2	0.0	-0.3	7.76	1.72

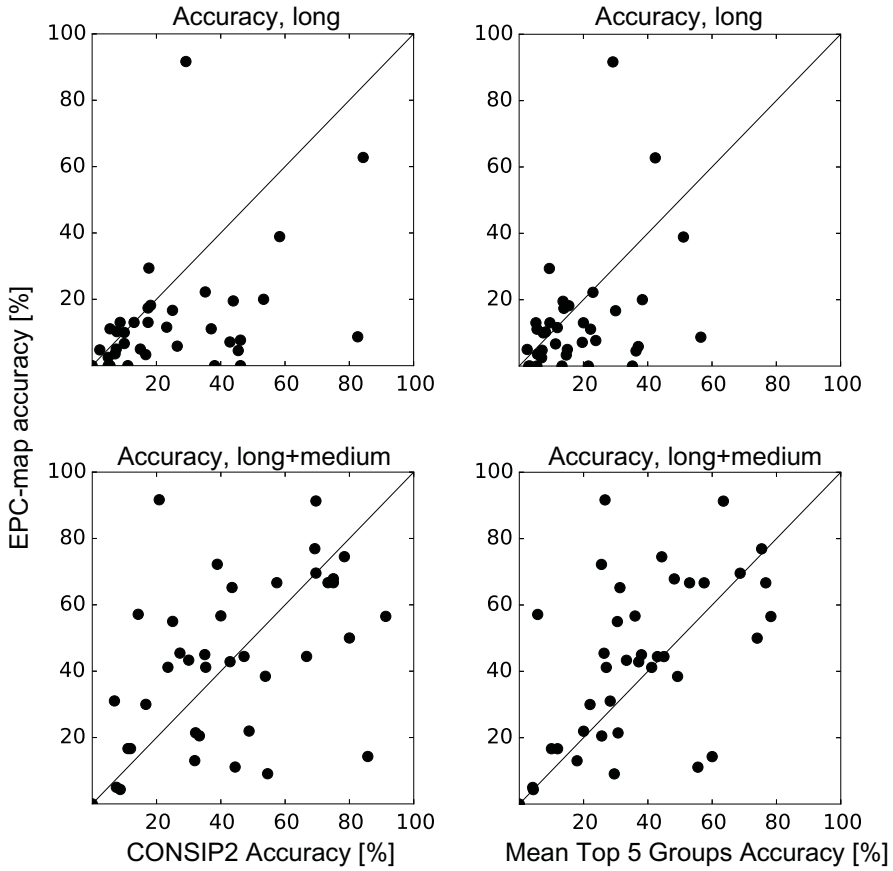

Fig. 6.5 Comparison of EPC-map in CASP11 with top methods. All results in this Figure refer to the top L/5 predicted contacts. All targets in this Figure are free-modeling targets. Comparison of long (sequence separation > 23) range and long+medium (sequence separation > 11) range contact prediction accuracy of EPC-map with CONSIP2, the winning method in CASP11. In addition, we compare our accuracy with the per-target mean accuracy of the top 5 contact predictors of CASP11. Figure adapted from: Mabrouk et al. [126].

is valuable for contact prediction. Note that no pure evolutionary method is within the top five contact prediction groups in CASP11 [33]. We view this as further evidence that the physicochemical information of EPC-map significantly contributed to our CASP11 performance. However, there is a clear gap in performance between our earlier experiments and the results in CASP11. This might be partially explained by the nature of the CASP11 data. Many proteins in CASP have complex topologies, high contact order, or are otherwise difficult to fold. EPC-map depends on decoys that are generated by *ab initio* structure prediction. Thus, if the decoys do not contain any native-like contact patterns, EPC-map will also fail to extract them. This might have been true in CASP11: The decoy quality was too low to enable effective contact prediction with EPC-map. Nevertheless, this is not

an inherent limitation of the algorithm. A decoy generation method that provides higher quality decoys will also enable EPC-map to predict higher quality contact maps. It would be interesting to test EPC-map on decoy ensembles from other algorithms that are designed to fold large proteins and/or complex protein topologies [23, 99]. Nevertheless, we would like to point out that our rather recent approach of EPC-map had very little time to mature, compared to the other participating sequence-based methods. Considering this fact and the high competitiveness of the CASP experiment, this is a very encouraging result.

6.4.2 The Impact of EPC-Map in CASP11 Free-Modeling

In this section, we analyze the free-modeling results of RBO Aleph in CASP11 with special attention to the impact of EPC-map.

Overall, RBO Aleph submitted predictions for 40 out of 45 free-modeling domains. For the assessors' formula and first submitted model, RBO Aleph ranked first by average Z-score > 0 and ranked third by sum Z-score > -2. For the best-of-five models, RBO Aleph ranked fourth for by average Z-score > 0 and sevenths by sum Z-score > -2[4].

The Performance of RBO Aleph Is Mostly Attributed to *Ab Initio* Structure Prediction

Before we analyze the impact of EPC-map and guided-MBS on free-modeling performance, we first confirm whether the *ab initio* prediction pipeline is actually the main contributing factor to our CASP11 results. We analyze whether our successful models (Z-score > 0) stem from the template-based modeling or *ab initio* structure prediction branch of the pipeline. At the time of this study, we had access to 27 domains for which structures were available. RBO Aleph modeled 20 out of 27 domains in CASP11 using the *ab initio* modeling pipeline.

In total, 17/20 *ab initio* models had positive Z-scores (Figure 6.6). In contrast, only two of the seven domains that stem from the template-based modeling pipeline reached positive Z-scores. Therefore, the performance of RBO Aleph stems mostly from the *ab initio* part of the pipeline. The assessors of CASP9 and CASP10 noticed that most successful groups were not pure *ab initio* folders but instead found and utilized templates [102, 199]. In contrast to their findings, our results indicate that *ab initio* folding is able to deliver state-of-the-art performance in free-modeling. Since the performance of RBO Aleph comes from the *ab initio* pipeline, we further analyze the impact of the unique *ab initio* components, EPC-map and guided-MBS.

[4] In total, 44 automatic methods participated in the free-modeling category of CASP11.

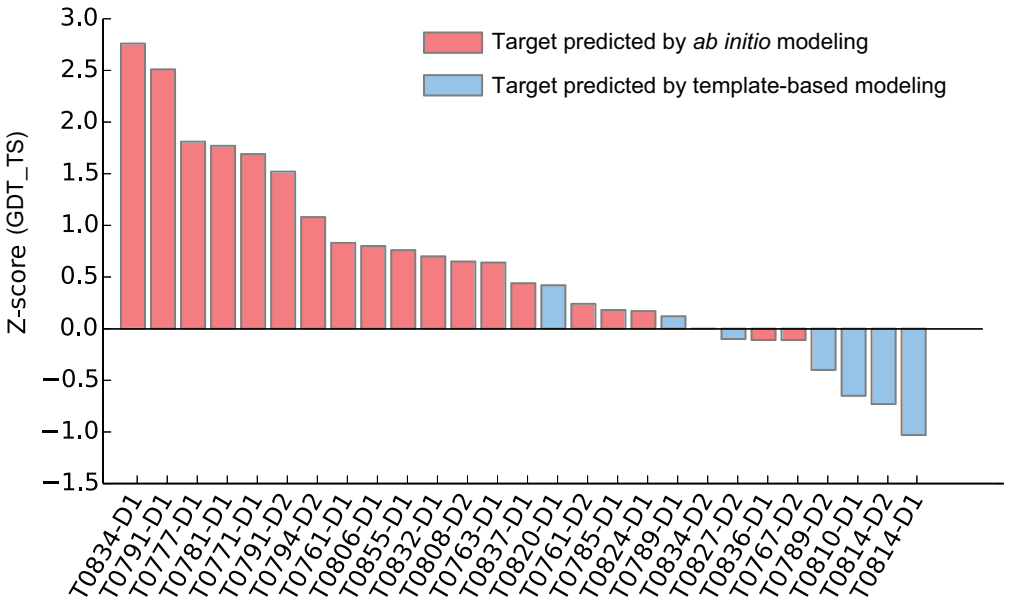

Fig. 6.6 Free-modeling Z-scores of RBO Aleph. Target domains that RBO Aleph predicted with the *ab initio* pipeline are shown in red, domains that are modeled with the template-based modeling pipeline in blue. Out of 27 analyzed domains, only seven are modeled with template-based modeling. The majority of *ab initio* targets have positive Z-scores, indicating above average performance of RBO Aleph for these targets. This demonstrates that the performance of RBO Aleph mainly stems from *ab initio* structure prediction. Figure source: Mabrouk et al. [126].

Evaluation of Conformational Search with Model-Based Search

Since the performance of RBO Aleph stems mostly from our *ab initio* pipeline, we investigate the components that likely contributed to the free-modeling performance: EPC-map and guided-MBS.

We first analyze the impact of the model-based search (MBS) conformational space search algorithm. MBS uses the distribution of decoys to incrementally build a model of the energy landscape. The algorithm uses this information to guide search towards low-energy regions (see section 6.2.1).

In this section, we analyze the performance of MBS conformational space search without the use of EPC-map contacts to evaluate the net effect of MBS as a search strategy. Ideally, we would compare our decoy sets with the decoy sets of other servers to assess the impact of our search method. Unfortunately, this information is not available to us. Thus, we compare MBS with the standard Monte Carlo sampling that is implemented in Rosetta 3.4 [170], which is considered to be a state-of-the-art sampling method. Note that this protocol is

Table 6.4 We used four times as many decoys for Rosetta than for MBS to provide similar computational resources to both methods. Table source: Mabrouk et al. [126].

Target	Domain	#MBS Decoys	#Rosetta Decoys
T0761	1	3000	12000
T0763	1	4000	16000
T0767	1	3000	12000
T0771	1	5000	20000
T0777	1	3000	12000
T0781	1	1000	4000
T0785	1	4000	16000
T0791	1	4000	16000
T0794	2	5000	20000
T0806	1	2000	8000
T0808	2	2000	8000
T0824	1	4000	16000
T0832	1	4000	16000
T0832	2	2000	8000
T0834	1	2000	8000
T0836	1	2000	8000
T0837	1	4000	16000
T0855	1	4000	16000

different to the BAKER-ROSETTASERVER method that participated in CASP11, which uses a more sophisticated protocol.

RBO Aleph varies the number of decoys in response to available computational resources and expected resources needed for a protein target (determined by protein size). We run Rosetta with approximately the same computational resources (four times more Rosetta decoys than MBS decoys, see Table 6.4). Note that Rosetta and MBS also use the same fragment library and energy function. Since the only difference in this comparison is the search method, this analysis should isolate the effect of MBS. Note that the following results refer to the domain definitions by RBO Aleph (Table 6.1).

In our analysis, we measure the median energy and median (best) GDT_TS of all generated decoys and the 10% lowest-energy decoys. This assesses whether MBS is a better optimizer of the energy, which is a prerequisite for improving search because lower energies should be associated with higher GDT_TS structures. The GDT_TS measures the structural accuracy of the decoys by assessing the deviation of the model to the native structure (see Appendix A.2.2).

For all domains, MBS finds lower-energy decoys (median energy difference across all ensembles between Rosetta and MBS is -35 Rosetta energy units, Figure 6.7). These findings confirm earlier studies that MBS effectively minimizes the energy [26].

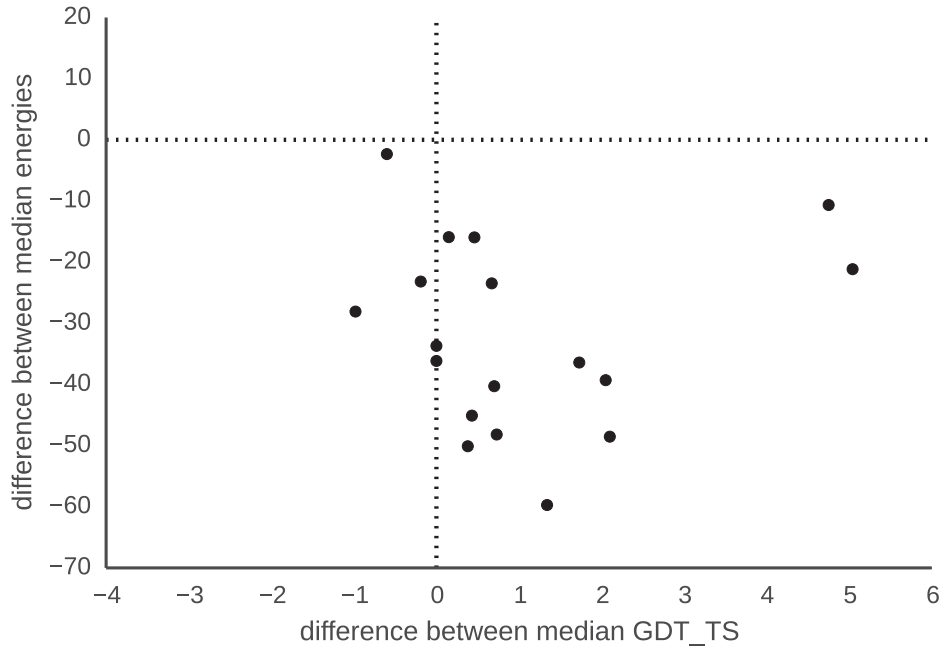

Fig. 6.7 Comparison of energies and GDT_TS of decoys obtained by MBS and Monte Carlo-based sampling strategy of Rosetta. The median energy and GDT_TS were calculated for each free-modeling domain using all decoys. Negative numbers at the y-axis indicate how much lower the energy of MBS-modeled decoys is compared with Rosetta-modeled decoys. The x-axis shows the difference of the median GDT_TS of MBS compared to Rosetta obtained decoys. MBS consistently finds lower energy decoys (-35 lower median energy over all in the ensembles) and better quality decoys (0.565 higher median over all decoys). This analysis is based on the RBO Aleph domain definitions. Figure source: Mabrouk et al. [126].

Compared to Rosetta decoys, the lower energy decoys of MBS translate into slightly higher GDT_TS (+0.565 median GDT_TS). MBS produces decoy ensembles with higher median GDT_TS than Rosetta Monte Carlo in 13/18 cases, but only in 9/18 cases for the 10% lowest-energy decoys. When comparing only the single highest GDT_TS decoy in the ensemble, without regarding the energy, we find that Rosetta finds at least one decoy with higher GDT_TS than MBS for all targets.

We interpret these results as evidence for deficiencies in the energy function, because lower energy decoys do not always translate to higher GDT_TS decoys. This issue might be further exacerbated by the exploitative behavior of MBS in energy landscapes of specific proteins that have a high degree of dissociation between energy and GDT_TS, or narrow funnels. MBS focuses only on a small number of funnels that will mislead search, if these funnels are not in the correct conformational space region.

To compensate for this issue, we combined EPC-map with MBS to provide an additional source of information. This additional information might help to steer search towards

biologically relevant regions that might not be identified by energy alone. In the next section, we investigate whether EPC-map contacts improve MBS predictions of CASP11 proteins.

Impact of EPC-map Contacts on MBS Decoy Quality

In the previous sections, we showed that the performance of RBO Aleph mainly stems from *ab initio* structure prediction and that MBS is a state-of-the-art search strategy. Since we use EPC-map contacts as input to guided-MBS, we wondered about the impact of predicted contacts on our performance. We re-run the MBS calculations without contacts on 20 target domains that we processed with *ab initio* structure prediction in CASP11 and compared the results with the RBO Aleph setup (EPC-map contacts as input for guided-MBS).

Figure 6.8 summarizes the results of this analysis. EPC-map-guided MBS calculations have higher energies than unguided calculations (median energy difference of 3.1/4.7 for all decoys/10% lowest-energy decoys)[5]. EPC-map-guided MBS only lowered the median ensemble energies in four out of 18 cases. However, contact guided-MBS slightly improves the median ensemble GDT_TS over unguided MBS calculations (median GDT_TS improvement of 0.45, Figure 6.8A). The improvement of the low-energy ensembles (best 10% in energy) is more pronounced (median GDT_TS improvement of 1.07, Figure 6.8B). This analysis demonstrates a small improvement on the GDT_TS when EPC-map contacts are used. Note that we assessed the quality of the generated decoys by the median GDT_TS of the ensemble, which is a conservative performance criterion.

We would like to point out some remarkable targets of this analysis. Target T0837 is a successful example of contact guidance in search (GDT_TS improves by 4.55; 3.94 for the low-energy ensemble). EPC-map also predicted precise contacts for T0806 but the contacts did not translate into higher quality decoys (median GDT_TS improvement: 1.27). The reason for this is probably the relatively large size of the target (256 residues) and the complex topology. However, there is no clear cut explanation whether contacts improved backbone quality of decoys in some cases, but not in others. The same holds true for targets with low contact prediction accuracy. Decoy quality decreases in some cases (T0761) and improves in others (T0777 and T0832_D1).

Impact of EPC-map Contacts on the RBO Aleph Pipeline

In the previous section, we showed that EPC-map contacts slightly increase the GDT_TS of MBS decoys. In this section, we study the effect of contacts on free-modeling predictions of RBO Aleph by rerunning our entire pipeline without contacts. This differs to the analysis in

[5]Precisely, median energy difference of the median energies of the investigated ensembles.

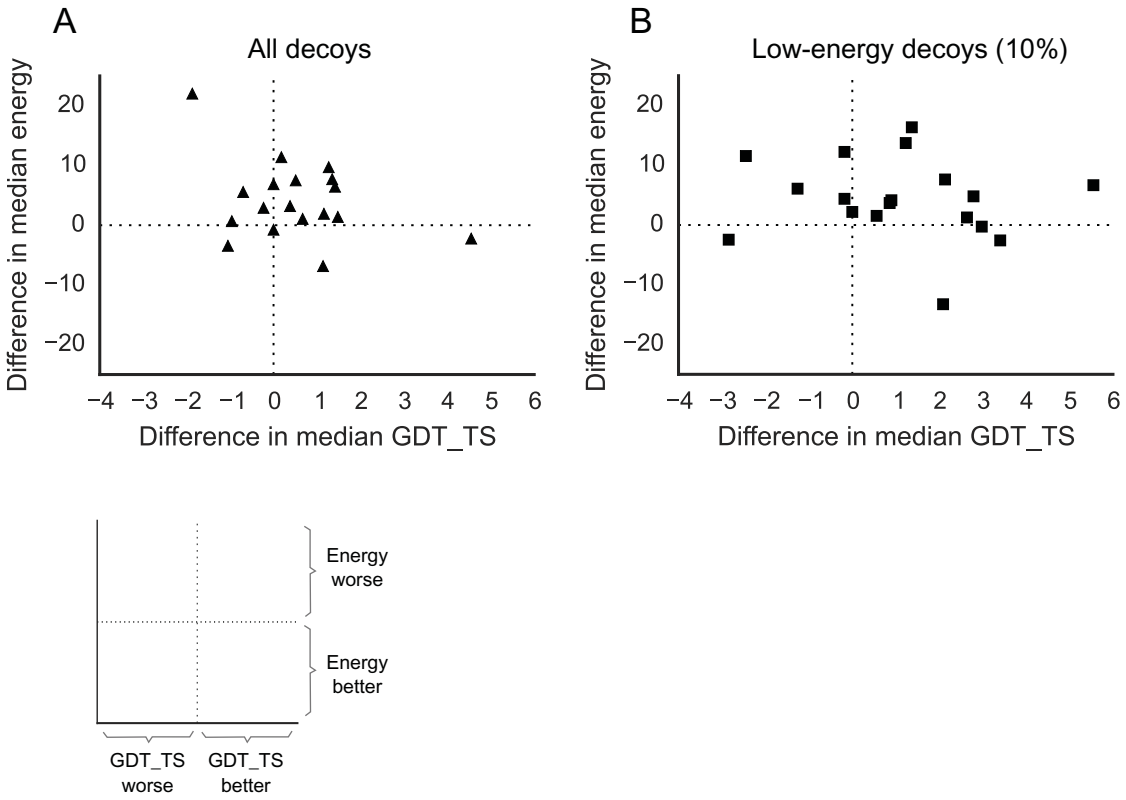

Fig. 6.8 Impact of EPC-map on MBS decoy quality. For this analysis, we run MBS without EPC-map contacts and compare the results to contact-guided MBS by the median GDT_TS in the decoy ensemble. **A**: Effect of EPC-map contacts on all MBS decoys. Overall, EPC-map slightly improves the decoy ensemble quality (median GDT_TS improvement of 0.45). **B**: Effect of EPC-map contacts on MBS decoys with 10% lowest energy. The improvement of the low-energy ensembles is more pronounced (median GDT_TS increase by 1.07). Contact-guided ensembles are not lower in energy. This might indicate that the contact information counteracts uncertain information in the energy function, that contact-guided decoys are not fully relaxed in the all-atom energy function, or a combination of both effects. This analysis is based on the RBO Aleph domain definitions. Figure adapted from: Mabrouk et al. [126].

the previous section (combination of contacts and search) because we now measure the net gain of EPC-map in the context of the entire pipeline (combination of contacts, search and decoy selection). In addition, we assess the pipeline performance on the official CASP11 domains.

We first compare the results to the other participating servers in CASP11. We compare the GDT_TS of the first model of RBO Aleph with: 1) the best submitted model 1, 2) the median GDT_TS of model 1 submissions of the top 5 servers (not including RBO Aleph), 3) the median GDT_TS of all model 1 submissions (Figure 6.9).

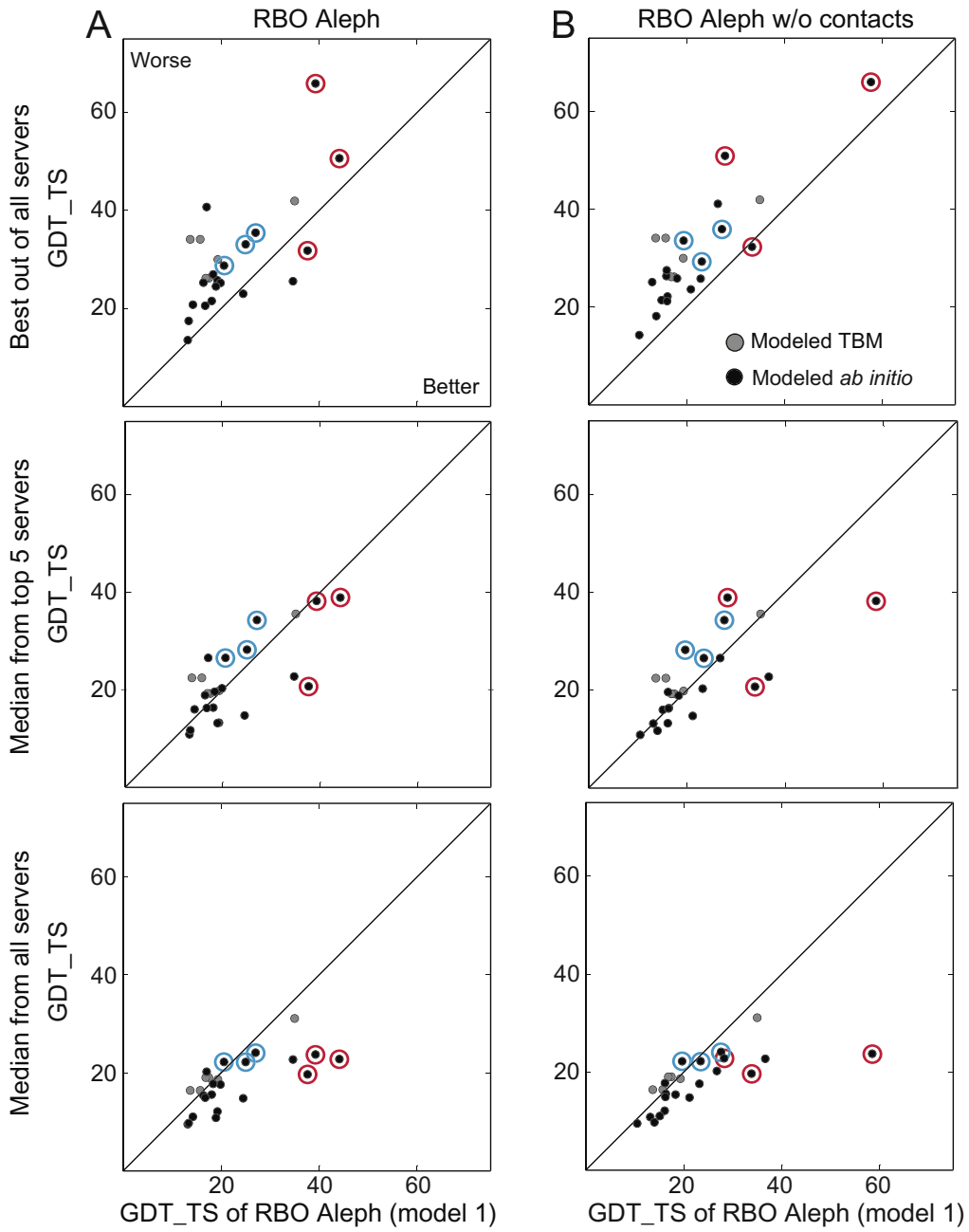

Fig. 6.9 Comparison of the GDT_TS of first models by our pipeline (with and without the use of contacts) to the first models submitted by other servers. We compare our first models to the best first model submitted by all servers, to the median GDT_TS of the first models submitted by top 5 servers in the FM category, and to the median GDT_TS of the first models submitted by all servers (excluding RBO Aleph in each analysis). FM targets that we modeled using ab initio are shown in black, targets we modeled using template-based modeling are shown in grey. The targets discussed in this thesis are highlighted by red (good predictions) and blue circles (bad predictions), respectively. Figure adapted from: Mabrouk et al. [126].

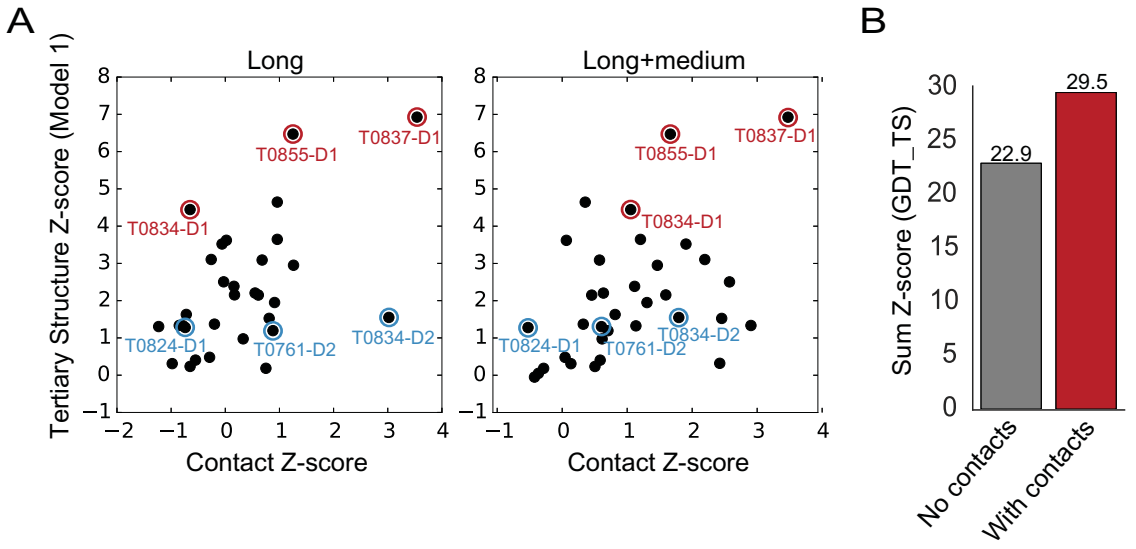

Fig. 6.10 Impact of EPC-map contacts on RBO Aleph free-modeling performance (model 1). **A**: Correlation between contact Z-scores and tertiary structure prediction Z-scores (model 1). The correlation Pearson correlation coefficient is 0.51 and 0.45 between long/long+medium contact Z-scores and tertiary structure Z-scores. This indicates that EPC-map at least partially contributed to the RBO Aleph CASP11 performance. Red circles indicate selected targets for which RBO Aleph submitted good models (high tertiary structure Z-scores). Blue circles point out targets for which RBO Aleph submitted poor models (low tertiary structure Z-scores). We discuss these targets in section 6.4.3. **B**: EPC-map contacts increase the sum Z-score from 22.9 to 29.5, an relative increase of 28.8% and an absolute increase by 6.6 points. Figure adapted from: Mabrouk et al. [126].

RBO Aleph submitted predictions with higher GDT_TS for 3/11/20 out of 27 domains compared to the best model 1 submissions, the median GDT_TS of top five servers, and the median GDT_TS of all submissions, respectively.

EPC-map contacts lead to higher performance (+21 sum GDT_TS of all model 1 submissions). In addition, the EPC-map increases the number of top predictions (rank 1) by RBO Aleph from one to three (Figure 6.9). Thus, EPC-map had a positive impact on our CASP11 free-modeling performance. Please note that RBO Aleph performs considerably well even without contacts (11/27 higher GDT_TS predictions than median model 1 submissions by top five servers).

We further investigated the impact of contact prediction on FM performance by analyzing the direct relationship between contact prediction performance and the outcome of tertiary structure prediction (Figure 6.10A). To do this, we analyzed the correlation between the contact prediction Z-score of EPC-map and the tertiary structure prediction Z-score of the first submitted model (model 1). The Z-scores of contact and tertiary structure prediction correlate moderately (Pearson/Spearman rank correlation coefficients are 0.51/0.47 and 0.45/0.48,

for long and medium+long range contacts, respectively). We view this as an indication that EPC-map contacts contributed, at least partially, to the free-modeling performance of RBO Aleph.

In our post-CASP11 analysis, we also recalculated the Z-scores of the first models that we computed without the use of EPC-map contacts (Figure 6.10B). This assesses the performance gain in terms of Z-scores, which ultimately determine the ranking of participating groups. The sum Z-score of EPC-map assisted RBO Aleph predictions is 29.5 and the sum Z-score of unassisted RBO Aleph is 22.9. Thus, EPC-map leads to a sum Z-score increase of 28.8% (relative increase) and an absolute increase by 6.6 points. Nevertheless, this does not affect the ranking because the next higher ranked server has a much higher sum Z-score (Zhang-Server [229], sum Z-score: 37.9) and the next lower ranked server a much lower score (MULTICOM-CLUSTER [38], sum Z-score: 20.9). However, EPC-map clearly increases the performance of the RBO Aleph pipeline in free-modeling.

6.4.3 Evaluation of Selected Free Modeling Targets

In this section, we highlight six selected targets for which RBO Aleph performed well or rather poorly. We highlight the targets with red (high Z-score) and blue (low Z-score) circles in Figures 6.9 and 6.10. We analyze the effect of decoy sampling (MBS), contact prediction (EPC-Map) and decoy selection (ProSA) on prediction quality. Whenever possible, we aim to link the performance to RBO Aleph to individual steps of our pipeline that succeeded/failed. However, we do not have access to the intermediate results of other servers, such as the decoy sets from their search methods. Therefore, we compare the median GDT_TS of our search method to the median GDT_TS of the best 20% server submissions.

High Quality Predictions

We first analyze three targets for which RBO-Aleph submitted models with high GDT_TS Z-scores (Figure 6.11).

T0834_D1: T0834 contains two domains and RBO Aleph incorrectly modeled the entire target as a single domain. For T0834_D1, RBO Aleph submitted a model with a GDT_TS of 37.9 (model 1). Our five submissions rank on positions 1-3 and 5-6 (model 3 of FFAS-3D ranks fourth). The target is comprised by three helices and a β-sheet motif. For this target, MBS+contacts generate decoy ensembles with a higher median GDT_TS (30.2) than all other serves except RaptorX-FM (GDT_TS: 32.07). EPC-map contacts have a long+medium range accuracy of 45.0% for this target. In addition, ProSA selects decoys within the top half

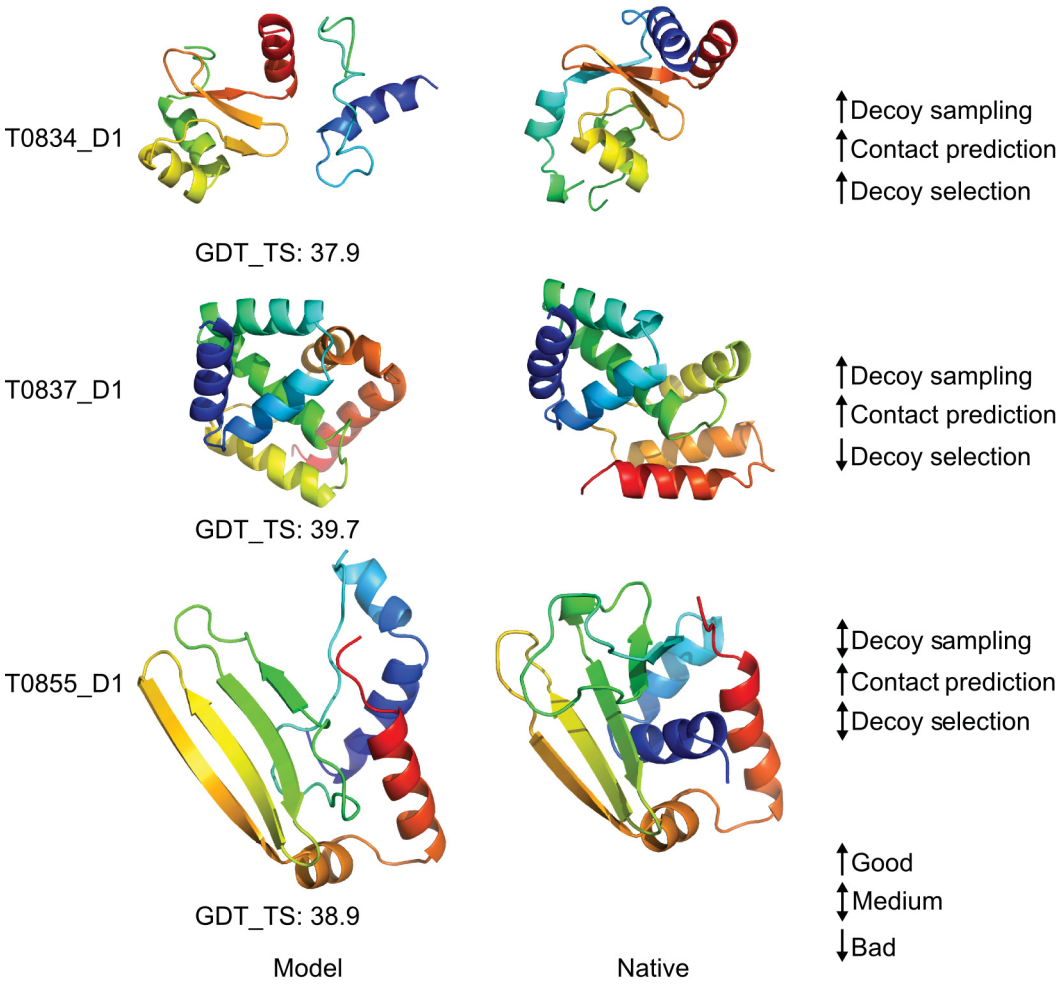

Fig. 6.11 High quality predictions produced by RBO Aleph. RBO Aleph submitted better models for the targets T0834_D1, T0837_D1, and T0855_D1 than most of the other servers. The arrows on the right side indicate the performance of the individual components (such as decoy, contacts prediction, and decoy selection). An upwards pointing arrow states good performance whereas a downwards pointing arrow indicates bad performance. Although the individual components did not always perform well, the overall prediction quality of the models is good. Figure source: Mabrouk et al. [126].

of the RBO Aleph decoy set. Despite incorrect domain parsing, the results for this target indicate that all components of RBO Aleph contributed to the modeling performance.

T0837_D1: Our model 1 for T0837_D1 ranks fourth with a GDT_TS of 39.7. The GDT_TS of the best (QUARK, GDT_TS: 65.7%) and second best model (RaptorX-FM, GDT_TS 45.45) are higher than the GDT_TS of RBO Aleph. The winning model (QUARK) successfully predicted the correct packing of the seven helices of T0837. EPC-map predicts precise contacts for this target (long+medium range contact accuracy: 91.7%) and our decoy set quality (GDT_TS: 43.6) is higher than the decoys of the top 20% submissions (40.2). The best decoy in the RBO Aleph decoy set has comparable quality to the QUARK submission (RBO Aleph, GDT_TS: 64.9, QUARK, GDT_TS: 65.7), but is not selected by ProSA. Overall, contact prediction and decoy sampling performed well for this target, but our performance is decreased by poor decoy selection. However, we still submitted one of the top five models for this target.

T0855_D1: Our model for T0855_D1 ranked seventh with a GDT_TS of 38.9. Again, contact prediction accuracy was high ($L/5$ accuracy 91.3%) but did not increase the quality of the sampled decoys (median GDT_TS without/with contacts on all decoys: 37.4/36.3; median GDT_TS without/with contacts on top 20% decoys: 41.0/41.7). Overall, the decoys of the top 20% servers were slightly better (GDT_TS 37.8). ProSA did not select any models from the top 20% sampled decoys. Overall, RBO Aleph predicted good contacts for this target but this is counterbalanced by modest decoy sampling and selection performance.

Low Quality Predictions

In this section, we analyze three targets for which RBO Aleph submitted predictions with low GDT_TS and Z-scores (Figure 6.12).

T0761_D2: T0761_D2 is a two-domain target that RBO Aleph modeled as a single domain (Table 6.1). The RBO Aleph prediction for T0761_D2 is poorly packed. Contact prediction accuracy was low for this target (13.0% for long-range contacts). This probably affected MBS, which sampled decoys with a median GDT_TS of 24.5, which is lower than the top 20% decoys of other servers (median GDT_TS: 26.3). ProSA selects a decoy with a GDT_TS of 26.3. The two best model 1 predictions originate from template-based servers (SAM-T08-server, ZHOU-SPARKS-X) and the best model also used a template (BAKER-ROSETTASERVER, GDT_TS: 38.27). This indicates that RBO Aleph should have decided to use templates for this target. However, the best decoy found by MBS has a comparable

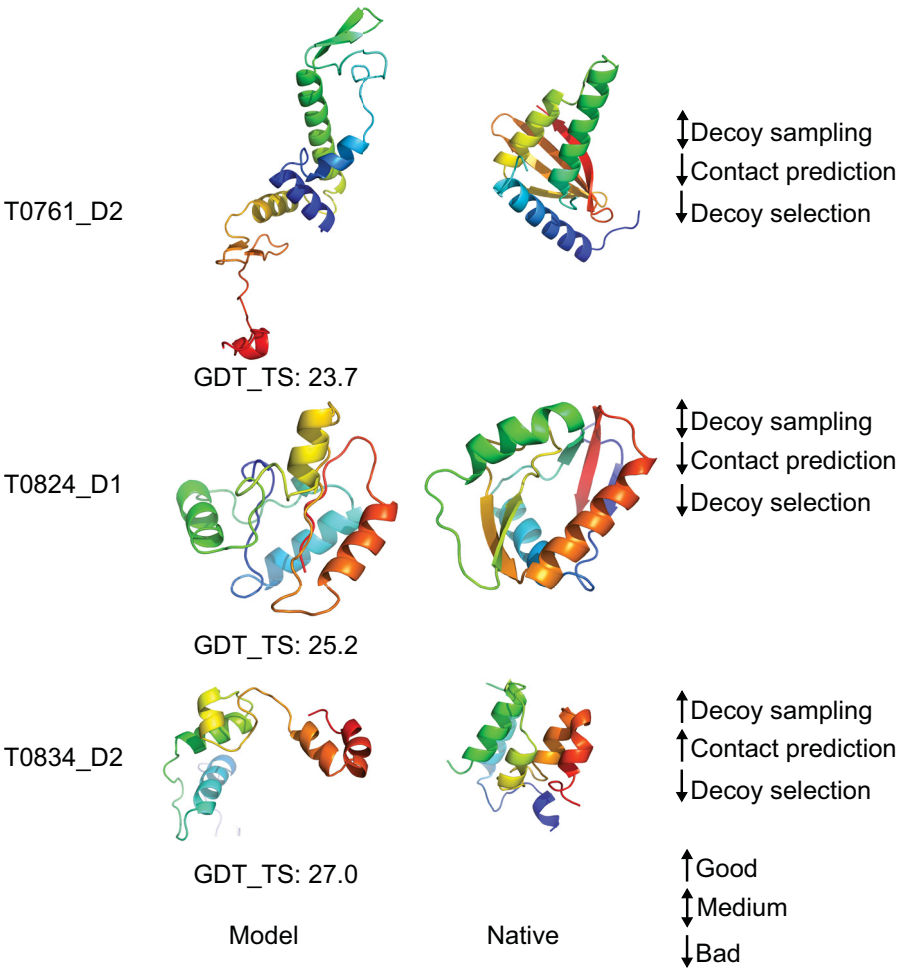

Fig. 6.12 Low quality predictions produced by RBO Aleph. RBO Aleph has submitted worse models for the targets T0761_D2, T0824_D1 and T0834_D2 than most of the other servers. The arrows on the right side indicate the performance of the individual components such as decoy sampling, contacts prediction and decoy selection. An upwards pointing arrow states good performance whereas a downwards pointing arrow indicates bad performance. Although the individual components sometimes performed well, the overall prediction quality of the models is weak. Figure source: Mabrouk et al. [126].

GDT_TS (35.8), which suggests that pure *ab initio* methods can be competitive to template-based modeling for difficult targets.

T0824_D1: For this target, none of our components performed well. Contact prediction was poor with a long+medium range accuracy of 9.1%. MBS also sampled lower GDT_TS decoys than the median decoys of the top 20% servers (GDT_TS of 22.6 and 27.6, respectively). ProSA did not select a decoy within the top 20% best decoys. This leads to a wrong predicted helix between residues 65 and 71 and a missing β-sheet motif in the core region of the structure. However, the best GDT_TS of this target is also rather poor (33.0, MULTICOM-CONSTRUCT), indicating that T0824_D1 was challenging for all predictors.

T0834_D2: T0834 is a two domain protein with a non-sequential first domain. RBO Aleph modeled the full-length target as a single domain. RBO Aleph modeled the first domain quite well (see above) but failed to model the second domain. For T0834_D2, decoy selection failed which resulted in a poor prediction by RBO Aleph. The contact prediction performance (long-range accuracy: 29.4%) and decoy sampling performance are reasonable for this target (median GDT_TS: 31.1, median 20% top groups: 30.8). However, ProSA selects a model that is worse than a randomly selected model (GDT_TS of 27.0 for ProSA).

One explanation for the failure on T0834_D2 is that domain parsing failed to split the domains, which influences all subsequent modeling tasks. This targets highlights that a weak performing component diminishes the performance of the entire pipeline, even if the other components work reasonably well.

Summary of Single Targets Analysis

The detailed analysis of individual targets provides some insight to the successes and failures of our server. The most contributing factor to tertiary structure prediction success is the sampling of high GDT_TS decoys. This is influenced by our conformational space search strategy and quality of the contacts that we use for guidance. Furthermore, decoy selection can be decisive for prediction success, which is clearly demonstrated by T0834_D2. For this target, we predict highly accurate contacts and high GDT_TS decoys. However, decoy selection fails to select models with good GDT_TS.

This points out that the performance of the individual components is insufficient to explain the performance of a structure prediction pipeline. Thus, we analyze the effect of different component combinations on the pipeline level, and investigate how they interact, in the next section.

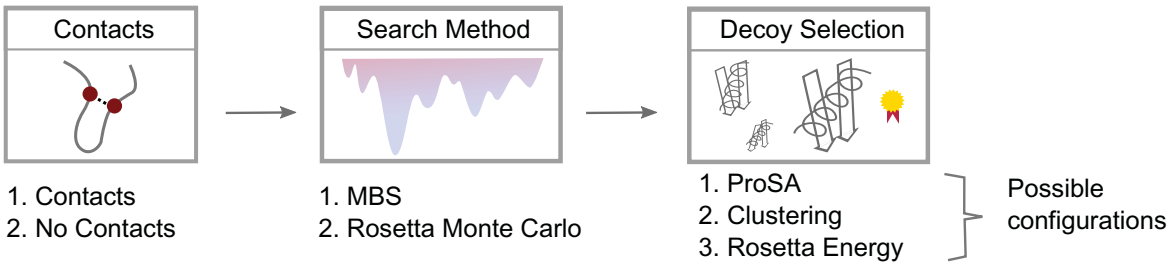

Fig. 6.13 Overview of the analyzed component configurations of RBO Aleph. We test several component combinations of RBO Aleph and their impact on the final prediction outcome of the pipeline. We tested three different components by removing them or exchanging them with another method: Contact prediction, search method, and decoy selection. We compare the following configurations: No contacts/contacts (EPC-map), MBS/Monte Carlo (Rosetta 3.4) as search methods, and ProSA/Clustering/Rosetta Energy for decoy selection.

6.4.4 Pipeline-Level Analysis of RBO Aleph

In the previous section, we analyzed the contribution of EPC-map and guided-MBS to the performance of RBO Aleph. We found that EPC-map increases the decoy quality of MBS and that EPC-map improves the outcome of the pipeline. However, we also found evidence that the pipeline components might interact in unexpected ways and that poor performance of a single component can diminish the performance of the entire pipeline. In this section, we investigate this aspect further.

To quantify the net effect of the individual components on the entire pipeline, we set out to systematically investigate the interrelation between individual components. We rerun all predictions using different pipeline configurations and measure the pipeline quality by the GDT_TS of the first and best-of-five selected final models. Since we analyze the *ab initio* pipeline of RBO Aleph, we are interested in the following questions: Do contact constraints influence pipeline performance? Does MBS as a search method improve pipeline performance? Which decoy selection method performs the best?

Recall that RBO Aleph uses the following combination of components: Contacts from EPC-map, MBS as a search method, and ProSA for decoy selection. The RBO Aleph configuration is the baseline for our pipeline analysis. Figure 6.13 lists the possible method combinations in this analysis.

We first compare the relative performance of different component combinations by a head-to-head bootstrap analysis. We then investigate whether a different component combination would have changed our rank in CASP11. Lastly, we discuss the pipeline analysis results.

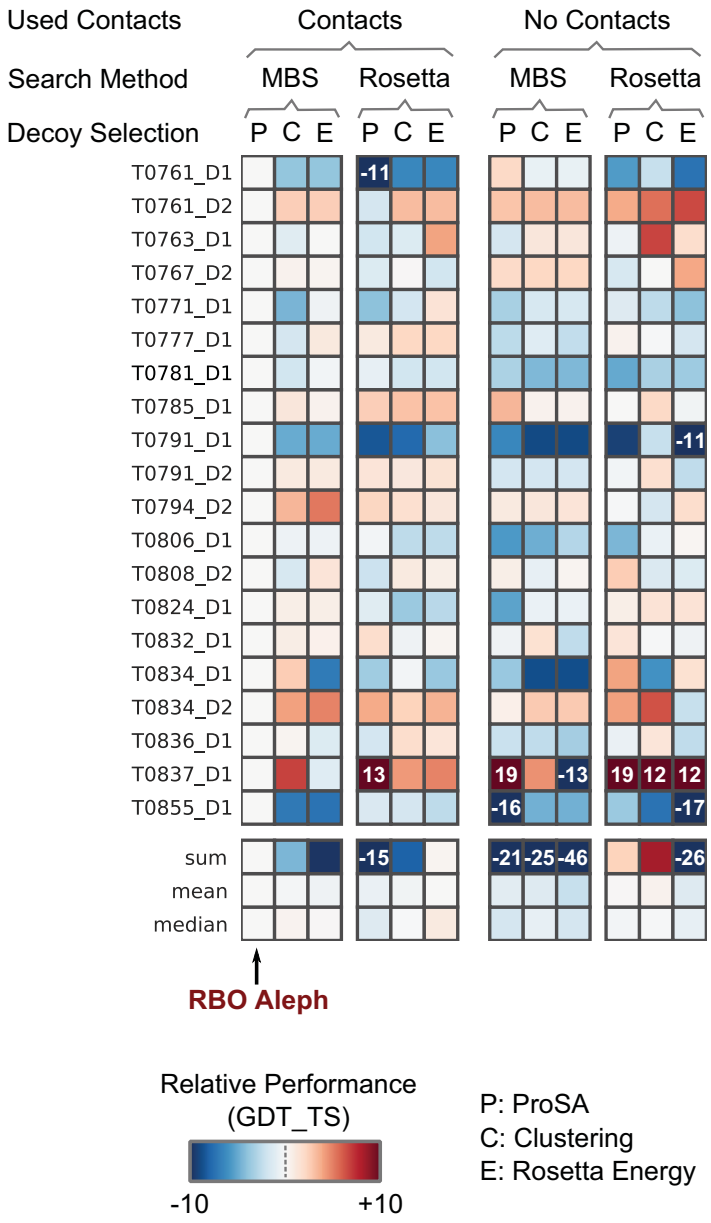

Fig. 6.14 Target-based analysis of various component combinations. The color of the heat map cells encode the performance difference of a particular component combination to RBO Aleph. Results based on model 1 and official CASP11 domain assignments. Figure adapted from: Mabrouk et al. [126].

Head-To-Head Bootstrap Analysis Reveals Effective Component Combinations

We are only able to test a small number of domains (20 official FM domains, see section 6.3.6). In addition, the difference in performance is generally small and often attributed to a small number of domains that benefit from a particular pipeline configuration (Figure 6.14). Therefore, the outcome of the analysis might be affected by a slightly different target composition. To account for this effect, we perform a head-to-head bootstrap analysis. In this analysis, we consider two RBO Aleph configurations and randomly remove five target domains. The pipeline with the higher mean GDT_TS is the "winner" of this comparison. We repeat this process 1000 times for each pair of combinations. The fraction of "wins" indicates the relative performance between two methods over different target compositions.

Figure 6.15 shows the result of this analysis. The head-to-head bootstrap analysis confirms that RBO Aleph is an effective component combination. Nevertheless, we find other component combinations that perform as well as RBO Aleph. Notably, some component combinations even perform slightly better: No Contacts/Rosetta/ProSA (RBO Aleph fraction of wins: 0.36) and No Contacts/Rosetta/Clustering (RBO Aleph fraction of wins: 0.25). We now highlight two interesting findings of our pipeline analysis.

RBO Aleph Is an Effective Contact Guided Configuration Considering all contact guided configurations, we find that RBO Aleph performs as good or better (measured by GDT_TS of the pipeline output) than all other contact-guided method combinations. It should be noted that differences in performance are generally quite small. Among the contact guided combinations, Rosetta with ProSA performs worst (sum/mean/median ΔGDT_TS of -14.7/-0.7/-1.3).

The Pipeline Analysis Detects Other Effective Component Combinations We also remove EPC-map contacts from the pipeline and test the performance of unguided configurations. MBS clearly benefits from contact constraints (sum/mean/median ΔGDT_TS of -21.2/-1.1/-1.8 for MBS without contacts with ProSA). Interestingly, Rosetta Monte Carlo without contacts and either ProSA (sum/mean/median ΔGDT_TS of 2.3/0.1/-0.3) or Clustering (sum/mean/median ΔGDT_TS of 8.4/0.4/-0.1) performs slightly better than the RBO Aleph configuration. Rosetta does not perform better if we select decoys by Rosetta Energy.

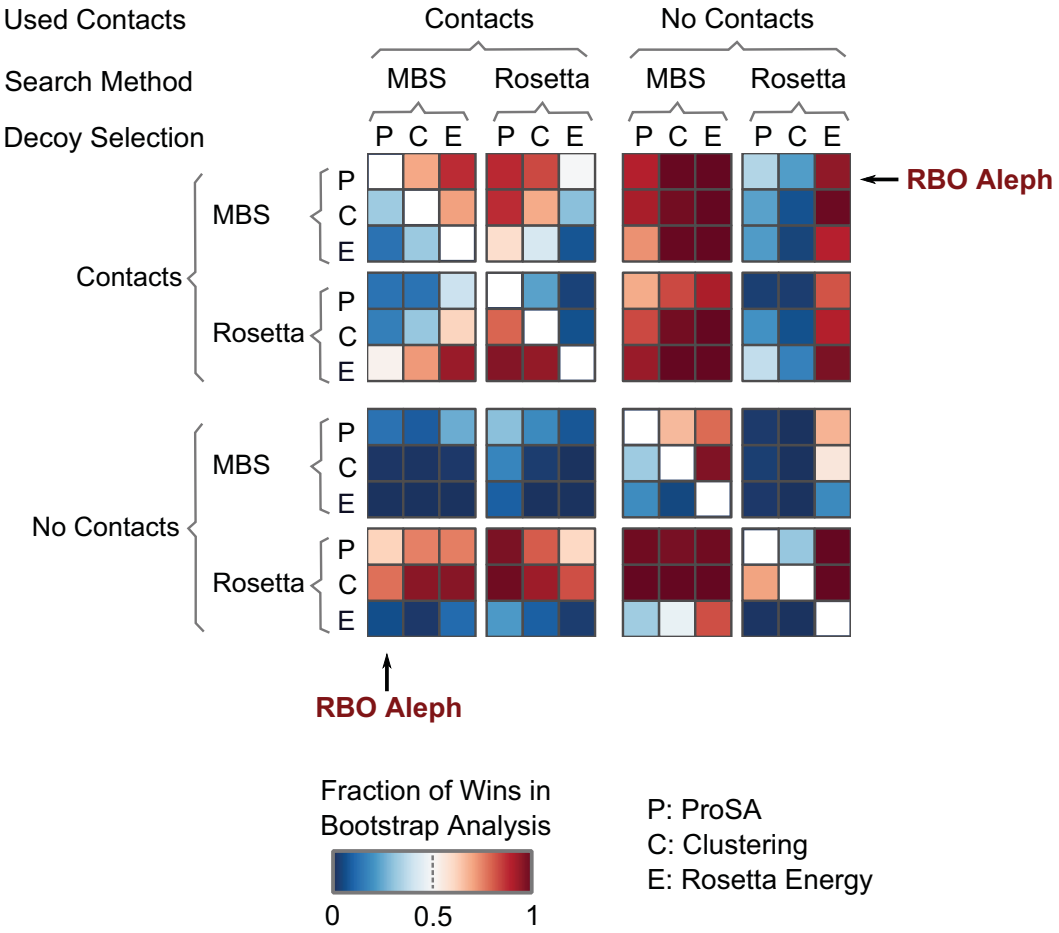

Fig. 6.15 Head-to-head bootstrap analysis of the different component combinations. In this analysis, we randomly remove five target domains for a pair of component combinations. The pipeline with higher mean GDT_TS scores a "win". We repeat this process 1000 times for each component combination pair. The colors of the heat map cells encode the fraction of "wins" of a particular method combination over another one. Figure adapted from: Mabrouk et al. [126].

Impact of Component Combinations on CASP11 Ranking

Would RBO Aleph ranked differently if we would have used a different pipeline combination in CASP11? We set out to answer this question by a rank stability analysis. Because the rank might be an effect of a particular target composition, we randomly remove five target domains from all servers and compute the resulting rank by the sum of Z-score (based on GDT_TS) of the first model (model 1). We repeat this analysis 1000 times and report statistics of this rank analysis. We analyzed official 27 target domains (including TBM domains). Please note that this is about half of the official target domains (45) and therefore the outcome of the analysis might be slightly different for the entire target set.

Table 6.5 Impact of component combinations on CASP11 ranking of RBO Aleph for the first model. For each component combination, we randomly remove five target domains and compute the rank among server groups on FM targets by the sum of Z-scores (GDT_TS, model 1). We repeat this process 1000 times and report the mean, best, and worst rank. MBS without contacts results in a lower rank than RBO Aleph. This indicates that contact constraints benefit MBS calculations. No Contacts/Rosetta and Energy for decoy selection also ranks lower. For all other methods, the rank is remarkably stable. Table source: Mabrouk et al. [126].

Used Contacts	Search Method	Decoy Selection	Mean Rank	Best Rank	Worst Rank
Contacts	MBS	ProSA	3.0	2	6
Contacts	MBS	Clustering	3.0	2	6
Contacts	MBS	Energy	3.0	2	6
Contacts	Rosetta	ProSA	3.1	3	8
Contacts	Rosetta	Clustering	3.0	2	7
Contacts	Rosetta	Energy	3.0	2	5
No Contacts	MBS	ProSA	3.5	3	9
No Contacts	MBS	Clustering	4.1	2	9
No Contacts	MBS	Energy	6.0	3	10
No Contacts	Rosetta	ProSA	3.0	2	8
No Contacts	Rosetta	Clustering	3.0	2	4
No Contacts	Rosetta	Energy	3.8	3	10

Only four out of 12 component combinations impact ranking (see Table 6.5). Interestingly, this includes all configurations based on MBS without contacts (mean rank 3.5-6.0). The results suggest that, at least for MBS as a search method, the EPC-map contacts resulted in a higher CASP11 rank.

The rank of the method combinations is quite stable because the Z-score difference of RBO Aleph (22.9) to the next higher ranked and lower ranked servers is relatively high (Zhang-Server [229], sum z-score: 37.9; MULTICOM-CLUSTER [38], sum Z-score: 20.9). For the best out of five submitted models, Rosetta with EPC-map constraints and decoy selection by Rosetta Energy is the best combination (mean rank 5.0, see Table 6.6). However, we find again that EPC-map contacts slightly improve the ranking of the pipeline: The best component combination that uses contacts ranks 5.0 while the best combination that not uses contacts only ranks 5.5 (Rosetta with ProSA, mean rank over 1000 trials).

Table 6.6 Impact of component combination on CASP11 ranking of RBO Aleph for the best-of-five models. For each component combination, we randomly removed five target domains and computed the rank among server groups in CASP11 on FM targets by the sum of Z-scores (GDT_TS, best-of-five models). This process is repeated 1000 times and the mean, best and worst observed rank is reported. Table source: Mabrouk et al. [126].

Used Contacts	Search Method	Decoy Selection	Mean Rank	Best Rank	Worst Rank
Contacts	MBS	ProSA	7.6	6	13
Contacts	MBS	Clustering	7.9	6	13
Contacts	MBS	Energy	7.1	5	13
Contacts	Rosetta	ProSA	6.9	4	10
Contacts	Rosetta	Clustering	6.3	4	9
Contacts	Rosetta	Energy	5.0	4	9
No Contacts	MBS	ProSA	7.3	5	13
No Contacts	MBS	Clustering	9.7	7	16
No Contacts	MBS	Energy	10.8	8	17
No Contacts	Rosetta	ProSA	5.5	4	8
No Contacts	Rosetta	Clustering	6.1	4	10
No Contacts	Rosetta	Energy	7.0	4	10

6.4.5 Interpretation of the RBO Aleph Analysis

In this section, we discuss the four main findings of our pipeline analysis. Our main intention of the full pipeline analysis is to increase our understanding of the interaction between components in structure prediction. We investigated the performance RBO Aleph variations along three dimensions: Contact constraints, search method, and decoy selection.

RBO Aleph Leverages Contacts Effectively

Our first finding is that EPC-map improves the performance of MBS. There are several pieces of evidence that support this finding: The pipeline analysis (see Figure 6.15), the impact of EPC-map on MBS decoys quality (see Figure 6.8), and the correlation between contact and structure prediction Z-scores (see Figure 6.10).

MBS leverages sampled decoys to build an energy landscape model and focuses search in the most promising regions. This strategy generally succeeds in generating low energy decoys (Figure 6.7). The downside of this strategy is that inaccurate information leads to search in the wrong regions of the conformational space. This issue might be especially true for difficult modeling targets in CASP. Search in these cases might be easily mislead by local minima, narrow funnels, and errors in the energy function. EPC-map contacts provide

additional information about the native state. This guides MBS towards regions in search space that contain native-like conformations. Therefore, information in contacts counteracts the issue of overly exploiting inaccurate or wrong information about the energy landscape.

The Exploration-Exploitation Problem Is Not Solved in Contact-Guided Structure Prediction

Our second finding is that MBS clearly benefits from EPC-map contacts, while the impact of contacts on Monte Carlo search is much lower. On the other hand, the best-of-five performance is higher for Monte Carlo search. These results suggest that neither search method, MBS and Rosetta, optimally leverages EPC-map contacts. MBS has a tendency to strongly exploit contact information. This results in decoy ensembles with higher GDT_TS (see Figure 6.8). However, the focused search behavior also misses some good models that are found by Rosetta Monte Carlo (see Table 6.6).

Possibly, broad sampling with Rosetta Monte Carlo increases the probability of generating higher GDT_TS decoys, especially when coupled with an effective decoy selection method. For the extremely difficult FM targets in CASP11, this broad search behavior is maybe an advantage, especially if contact accuracy is low. Broad exploration of the energy landscape is much less biased by information that is potentially wrong. On the other hand, directed search methods such as MBS might benefit from a higher degree from future improvements in contact prediction. Since Rosetta is an explorative sampling method and MBS an exploitative search method, this indicates that the exploration-exploitation trade-off is unsolved in the context of contact-guided search.

Pipeline Analysis Reveals Other Effective Component Configurations

Our third finding is that rigorous pipeline analysis can reveal configurations that are as effective or better than RBO Aleph. Interestingly, we find that using Rosetta Monte Carlo without contacts and ProSA/Clustering performs slightly better in our experiments than RBO Aleph. Note that this does not necessarily mean that this configuration would have also performed better in CASP11, since the rank of other component combinations does not change and we were only able to analyze a subset of the official free-modeling domains (see section 6.3.6).

Component Combinations Interact in Unexpected Ways

Our fourth finding is possibly the most interesting: Certain method combinations seem to go well with each other and some do not. We find that especially decoy selection is sensitive towards the preceding pipeline components, which other studies confirm as well [104, 138].

This has important implications for the analysis of protein structure prediction systems. It seems that the analysis of individual components is not sufficient to extrapolate to the performance on the entire pipeline. If we would not consider some components in pipeline analysis, we would come to different, even contradicting, conclusions regarding the performance of RBO Aleph. For example, if we would only consider Clustering as a decoy selection method, we would have come to the conclusion that contact constraints improve MBS but hurt Rosetta. In contrast, the additional systematic testing of multiple decoy selection methods shows that this view would be overly simplified.

More importantly, the systematic analysis of the pipeline reveals that improvements of individual components do not always lead to improvements of the entire pipeline. This finding suggests that the components in structure prediction pipelines interact in unexpected, counter-intuitive ways.

This has important implications for the development of structure prediction methods. A practitioner cannot simply integrate a seemingly improved method into a pipeline, because it might interact negatively with other components, effectively reducing the performance of the entire system. We suspect that one reason for this behavior is that components of a protein structure prediction pipeline are finely tuned and even small changes of individual components therefore might have stark impact on pipeline performance. Until we understand these interactions, a detailed pipeline analysis may help to reveal potentially effective component combinations. A short term solution might be to release methods (such as decoy selection) not as fixed trained or optimized software packages, but to include re-training/re-parameterization routines that allow customization of the method to a new pipeline.

6.5 Conclusion

We presented an analysis of RBO Aleph which participated in the CASP11 contact prediction and free-modeling (FM) categories. In both categories, RBO Aleph ranked among the top five methods. To unravel the reasons for our success, we focused our analysis on components that are unique to RBO Aleph: Contact prediction with EPC-map and conformational space search with contact-guided-MBS.

EPC-map ranked second for medium+long-range contacts and fifth for long-range contacts. Considering that all other top methods in this category are sequence-based contact prediction methods that matured over many years, this is a very encouraging result.

In the FM category, RBO Aleph ranked first by average Z-score > 0 and third by sum Z-score > -2 (ranking based on first submitted models and assessors' formula). Our analysis provides evidence that MBS and EPC-map contributed significantly to our CASP11 performance. Interestingly, our results show that MBS significantly benefits from EPC-map contacts, while Rosetta's Monte Carlo sampling strategy does not. Nevertheless, Monte Carlo infrequently produces models with higher GDT_TS values. Since Monte Carlo relies heavily on exploration and MBS on exploitation, this finding indicates that balancing exploration and exploitation is in the context of contact-guided conformational space search is an unsolved problem. We believe that addressing this problem in future research would result in improved conformational space search strategies that would highly benefit from future developments in contact prediction.

Our most important finding, however, is that the individual components of a structure prediction pipeline interact in unexpected ways. An improved component does not necessarily lead to improved results of the entire pipeline. This finding is at least true for RBO Aleph, but might be also true for other structure prediction pipelines.

This finding has important implications: We speculate that component methods, developed in the context of a specific pipeline, might render ineffective when transferred to a different pipeline. If this is the case, this points to a significant obstacle to the community effort to advance structure prediction. Understanding these interactions of components, or at least ensuring compatibility between them, might be vital to advance protein structure prediction.

Chapter 7

Protein Structure Determination by Mass Spectrometry and Computational Biology

This work is based on the following publication:

Belsom, A.*, Schneider, M.*, Fischer, L., Brock, O., and Rappsilber, J. (2015). Serum Albumin Domain Structures in Human Blood Serum by Mass Spectrometry and Computational Biology. Mol. Cell. Proteomics, in print: mcp.M115.048504.

*contributed equally

Own contributions: I conceived and designed experiments. In particular, I conceived and designed the cross-link-guided conformational space search protocol. I implemented the cross-link-guided conformational space search protocol and performed all protein modeling experiments. I contributed analysis tools and analyzed the data, with a focus on the analysis of the structure determination results. I contributed to paper writing.

Contributions of co-authors: AB, LF, JR, OB conceived and designed experiments. AB performed experiments, in particular experimental cross-linking/mass spectrometry. LF contributed data analysis tools, in particular analysis of cross-link data. AB and LF analyzed data. AB, LF, JR, and OB contributed to paper writing. The following figures and tables were prepared in part or in modified form by co-authors of the original paper [B]: AB: Figure 7.1, 7.2, 7.3, 7.4, 7.5, 7.8, 7.9. LF: Figure 7.6, 7.7.

Extensions: This chapter contains no extensions of the original paper.

7.1 Introduction

In this chapter, we demonstrate an approach to structure determination by combining information from cross-linking/mass spectrometry (CLMS) experiments with computational protein structure prediction. This work follows the overarching theme of this thesis of leveraging novel information sources for protein structure prediction. Unlike the other information sources of this thesis, the information from CLMS is of experimental origin i.e. from a physical measurement on the specific protein.

Proteins exist in a crowded, cellular environment and constantly interact with other macromolecules. These interactions substantially influence the structure and conformation dynamics of proteins. Paradoxically, most of our knowledge stems from studying protein structures in isolation. Direct structure analysis in natural environments with experimental methods is extremely difficult. Therefore experimenters typically remove the protein from its native environment. However, removing a protein from its natural context might significantly alter the conformation. In addition, traditional methods (NMR, X-ray, EM) can only capture the influence of *in vivo* factors on protein structure to a very limited degree.

Only few experimental methods are able to obtain structural information from *in vivo* samples (refer to section 3.2.2 for an overview). For example, Sakakibara et al. [173] developed an in-cell NMR approach based on non-linear sampling that is able to analyze the structure of small proteins in living cells.

Another recent study employs genetically encoded photo-activatable cross-linkers to study ligand binding to CRF Class B GPCRs [40]. This study demonstrates that cross-linking/mass spectrometry is able to provide structural details under native conditions. However, the CLMS approach usually does not produce enough cross-links for *ab initio* determination of protein structure [91]. The method presented in this chapter overcomes this limitation.

We present a novel high-density CLMS hybrid method for *ab initio* structure analysis. We advance the state of the art by two key contributions: 1) We dramatically increase the cross-link density by using a highly reactive chemical as a cross-linking agent, the heterobifunctional chemical cross-linker sulfosuccinimidyl 4,4'-azipentanoate, sulfo-SDA [70]. 2) We combine the high-density CLMS data with guided model-based search (guided-MBS, see chapter 6). This extends the capabilities of guided-MBS to perform CLMS-driven conformational space search.

The synergy of high-density CLMS data and conformational space search enables the *ab initio* reconstruction of the human serum albumin domains (HSA) with good backbone accuracy (RMSD of 2.5/4.9/2.9 Å for the best-of-five structures of domains A/B/C). However, the most substantial impact of our method comes from its ability to probe protein structure under native conditions. We reconstruct the domain structures from in-serum HSA samples

with an RMSD of 3.5/5.2/3.8 Å for the best-of-five structures of domains A/B/C. Our experiments provide strong evidence that high-density CLMS-based hybrid methods enable detailed structure analysis of proteins in their natural environment.

Section 7.2 provides a general overview of the proposed hybrid method. Section 7.3 describes the method in detail, with a focus on the computational structure determination protocol, which is the main contribution of this dissertation. In section 7.4, we present the high-density CLMS data. We discuss the structural information in the data and current limitations. We then show that the density of our CLMS approach passes an important threshold: The *ab initio* reconstruction of protein structure. Section 7.5 concludes this chapter.

7.2 Overview of the Method

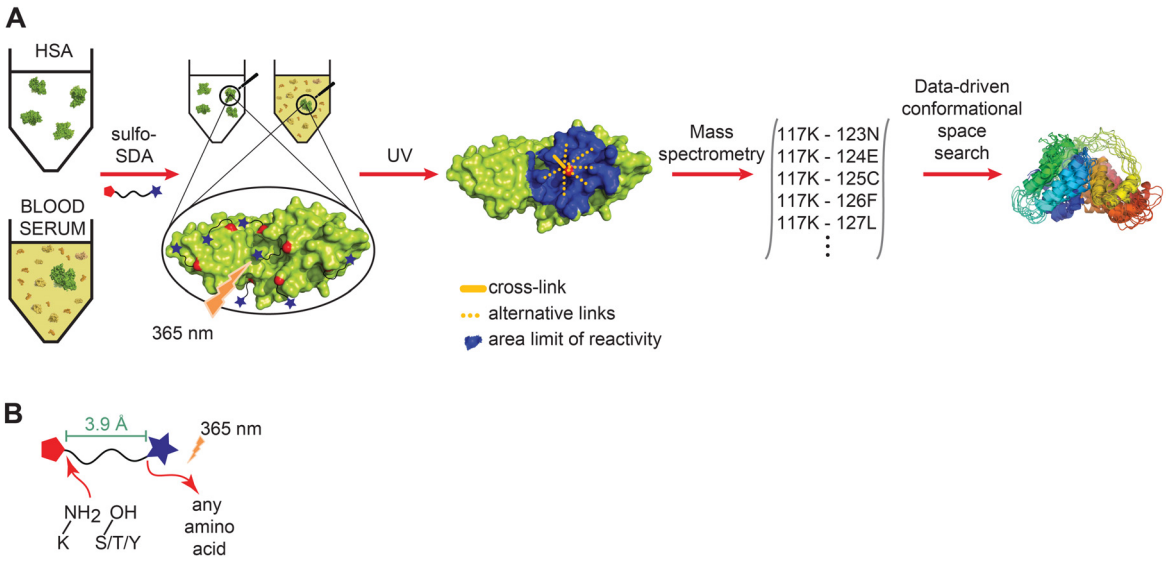

Fig. 7.1 Workflow of photo-cross-linking/mass spectrometry combined with computational conformational space search. **A**: We cross-link purified HSA and HSA from human blood serum using photo-reactive sulfo-SDA in a two-step procedure. Proteins are first decorated by the cross-linker at Lys, Ser, Thr, Tyr and N-terminus. Upon UV activation, the cross-linker links these residues to a nearby residue. The cross-linked protein is then subjected to a proteomic workflow, consisting of trypsin-digestion, liquid chromatography-mass spectrometry and database searching to identify the cross-linked residues. These intramolecular proximities are then used as experimental constraints during computational conformational space search. **B**: Schematic view of the cross-linker. Sulfo-SDA carries a specific NHS-ester on one and a diazirine group on the other end that can react with any amino acid. Figure source: Belsom et al. [10].

In this section, we give an overview of the proposed high-density CLMS-driven hybrid method.

The basic workflow of the proposed method is composed into two parts: Chemical cross-linking/mass spectrometric analysis of the protein of interest and conformational space search with CLMS data (see Figure 7.1). We cross-link purified HSA and human blood serum with sulfo-SDA in two steps. First, we label the protein by sulfo-SDA in the dark. Then, we expose the labeled protein to UV light, which triggers the formation of a highly reactive carbene species which immediately reacts with spatially close amino acids. We purify the cross-linked protein using SDS-Page. Note that SDS page exposes the protein to a denaturing condition. However, the structural information is already captured by the formation of cross-links, thus we do not need to worry about the integrity of the protein at this step. We excise the HSA band, digest the protein with a protease, and analyze the peptide mixture by LC-MS using a high-high acquisition strategy. The reduced selectivity of the photo-reactive diazirine on sulfo-SDA, increases the number of observed distance constraints significantly.

In the computational step, we use the CLMS data to guide conformational space search. First, the cross-linking data is combined with predicted contacts from EPC-map (see chapter 5). If we find a predicted contact that is also cross-linked, we view this as additional evidence that this contact might be formed and discard the cross-link in favor of the contact, which forms a tighter constraint. Since HSA is too big for *ab initio* structure prediction, we parse the structure into three domains. We then perform conformational space search with guided-MBS, a modified version of MBS [26] introduced in section 6.2.1. This modified version uses contacts and CLMS constraints to steer conformational space search in the low-resolution sampling phase, and to construct the funnel model. However, the assessment of funnel quality uses Rosetta's high-resolution energy function. This decoupling of search and funnel selection deals with noise in contact and CLMS constraints. In the last step, we select the representative model from the structure ensemble using a combination of Rosetta energy, a knowledge-based energy function, and the fit of the model to CLMS data.

7.2.1 Increasing Cross-Link Density with a Diazirine Cross-Linker

Current methods rely on selective cross-linking reagents, such as Bis(sulfosuccinimidyl)-suberate (BS3). Because of the high selectivity, CLMS data from this linker produces only few cross-links CLMS. Earlier studies show that this information is not sufficient for *ab initio* modeling of protein structure [91].

We hypothesize that the density of CLMS data can be increased by cross-linking reagents with less specific reactivity. If we increase the cross-link density, we might provide enough

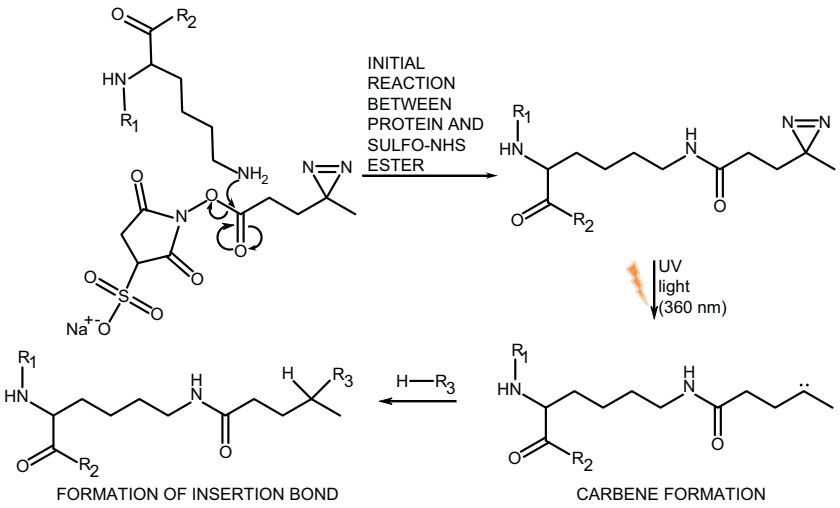

Fig. 7.2 Reaction scheme of sulfo-SDA. The NHS reacts with primary amines (-NH2) of the protein that exist at the N-terminus and in the side-chain of lysines. The NHS-ester also reacts to a lesser extent with serine, threonine, and tyrosine. In a second step, UV light at 360 nm activates the diazirine group. The resulting carbene species reacts with any other amino acid by forming a covalent bond with nucleophilic or active hydrogen groups. Figure source: Belsom et al. [10].

information for *ab initio* structure determination. The key element of our method is the use of sulfo-SDA as a cross-linking reagent. Sulfo-SDA is less specific than standard homobifunctional NHS-ester based cross-linker reagents. Sulfo-SDA carries two functional groups: A traditional NHS-ester on one end and a UV photoactivatable diazirine on the other end (Figure 7.2). The NHS-ester reacts with the side chains of lysines and to the N-terminus of the protein, to a lesser extent also with serine, threonine, and tyrosine. The diazirine is stable under normal conditions but UV light activation (320-370 nm) generates a highly reactive carbene [15]. Carbenes are highly promiscuous and react within femtoseconds with any organic molecules [212]. Thus, sulfo-SDA reacts with K/S/T/Y on one side and with any amino acid on the other side. This broadening in specificity increases the number of cross-links that we can produce and identify.

7.2.2 Mass Spectrometry and Data Analysis

The mass spectrometric detection and data analysis follows the experimental procedures that we introduced in chapter 4 (section 4.3). We spray the digested peptides onto a liquid chromatography column. The column separates the peptides by hydrophobicity. We gradually elute the peptides and spray them into the mass spectrometer. This step reduces the chemical complexity of the sample that needs to be analyzed at a time by the mass spectrometer.

We search raw mass spectrometry peak lists against a database containing the HSA sequence. We estimate the false discovery rate with a modified target-decoy search [60, 128]. Note that in this false discovery rate (FDR) analysis, we need to accept a certain amount of false positive cross-link matches (see 4.3.3 for an outline of this procedure). Thus, the final CLMS data will contain noise.

7.2.3 Leveraging High-Density CLMS Data for Protein Structure Determination

We now give an overview of the computational protocol that we use to leverage CLMS data.

Estimation of Upper Distance Bounds of Cross-Linked Residues

One challenge of modeling protein structure with CLMS constraints is that there is no straightforward way of converting a cross-link between two residues into a distance constraint. Multiple factors determine the upper distance bound of a cross-link, such as the length of the side chain and the cross-linker spacer length. Since the carbene of activated sulfo-SDA can react with any atom in any amino acid, we need to assume that the most distal atom of a side chain might have reacted. Long side chains, like lysine and arginine, approximately have C_α-Nz and N_{H2} distances of 6.0 Å and 6.6 Å, respectively [80]. If two lysines are linked, this would result in a C_α-C_α distance of 15.9 Å after adding the sulfo-SDA spacer length of 3.9 Å. However, we still need to account for conformational flexibility of the protein. Thus, we conservatively estimate 20 Å as an upper, Euclidean distance bound for the C_α-C_α atoms of cross-linked residues.

Conformational Space Search with CLMS Constraints

We add the CLMS constraints to the energy function using the modified lorentzian function that we introduced in chapter 5 (see section 5.3.8 on page 59).

Noise in cross-link data by overlength cross-links might arise from wrong database matches or errors in site assignment. Another source of overlength cross-links is the conformational flexibility of the cross-link in solution. A region of the protein might be cross-linked because the flexibility of the protein causes the region to be close at cross-link time, but apart in the crystal structure.

The Lorentz function adds an energy bonus if the constraint is satisfied. The energy bonus falls back to zero if the constraint is significantly violated, effectively ignoring the constraint. This accounts for the noisy nature of high-density CLMS constraints. We use

the guided-MBS algorithm for conformational space search, introduced in chapter 6 (see section 6.2.3 on page 86).

7.3 Experimental Procedures and Implementation

In this section, we give detailed information to our proposed structure determination method that is based on high-density cross-linking/mass spectrometry and conformational space search. Note that this dissertation focuses on the computational aspect of hybrid methods. Thus, we do not present the detailed experimental protocol of high-density CLMS that can be found in the original paper [10]. We focus on the computational aspect by describing the implementation of conformational space search in detail, including all pre-processing steps (domain splitting, contact prediction) and post-processing steps (model selection).

7.3.1 Implementation of CLMS-Driven Conformational Space Search

Domain Boundary Prediction

In protein structure prediction, the conformational search space grows exponentially with the protein length. This makes structure prediction of large protein structures (larger than 200 amino acids) challenging. In terms of structure prediction, HSA (PDB|1AO6) is a large protein with 576 residues. *Ab initio* structure prediction for proteins of this size is beyond the state of the art, even for the most advanced protocols. Thus, we decided to split HSA into its three domains to demonstrate the feasibility of our combined CLMS/search approach. We use the consensus of the following domain boundary predictors to split HSA into domains: DoBo [51], Threadom [228], and the domain boundary method implemented in PSIPRED [28]. We form the consensus by averaging the predicted boundary. Table 7.1 shows the individual predictions and the consensus.

Conformational Space Search with Realistic Energy Functions by Model-Based Search

We search the conformational space with cross-linking/mass spectrometry (CLMS) constraints with the modified model-based search algorithm, introduced in chapter 6, section 6.2.3. The guided-MBS algorithm in this chapter is very similar to the previously described algorithm. We here only provide the differences and parameters that we used for CLMS search.

First, we use CLMS constraints and contact constraints from EPC-map. Second, we generate 5000 samples per MBS stage. The third difference is the treatment of longer loop

Table 7.1 Domain boundary predictions by individual predictors. Table source: Belsom et al. [10].

Method	All predictions	Boundary A-B[a]	Boundary B-C[a]
DoBo	381/177/384/367/367/365/107/104[b]	177	384
Threadom	102/197/291/380/488[b]	197	380
PSIPRED	174/217/394[b]	217	394
Consensus		197[c]	386[c]

[a]Selected predictions, based on overall agreement between the methods. In case of ambiguous predictions (as for domain A), we went for the predictions with higher overall agreement in terms of residue deviation instead for the highest scoring predictions (174 with PSIPRED).

[b]Predictions are ranked by the corresponding method's score

[c]The consensus is computed by averaging the selected predictions.

sections. HSA contains long loops that are presumably rather flexible. Therefore, there will be a number of different loop conformations that are compatible with the native structure. However, the different compatible loop conformations might significantly influence the energy of the protein, leading to distortions in the energy landscape. Thus, we identify all loops longer than 15 amino acids and remove them from the energy calculation. We predict long loop regions with DISOPRED2 [217]. We still model these loops explicitly, but remove them from scoring with exception of the repulsive terms as described by [213]. We apply the same procedure to N/C-terminal residues that DISOPRED2 predicts to be disordered. This procedure leaves the following residues full scoring in the all atom phase: 2-71:115-194 for domain A, 200-262:308-381 for domain B and to 389-458:508-571 for domain C. Note that the RMSD calculated in this manuscripts does not consider the loop regions and is also calculated over these residues.

Contact Prediction

We augment the structural information of CLMS constraints with predicted contacts from EPC-map (see chapter 5 for a description of EPC-map). Note that the results of this chapter are computed with an early version of EPC-map that uses PSICOV [88] for computing evolutionary contacts. We replaced PSICOV with GREMLIN [96] in the current version of EPC-map.

Integration of CLMS Constraints and Predicted Contacts into Model-Based Search

We integrate CLMS constraints and EPC-map contacts with the modified Lorentzian function into guided-MBS (see section 5.3.8 for a detailed description of the function). We use the

following parameters for CLMS: l = 1.5 Å, u = 20 Å, and w=1.0. The choice of u = 20 Å is motivated of our estimation of the upper distance bound for sulfo-SDA CLMS constraints (section 7.2.3). For contacts, we use the following parameters: We use l = 1.5 Å, u = 8 Å, and w=1.0. Note that we define CLMS constraints between C_α atoms and contact constraints between C_β atoms (C_α for glycine).

We only used CLMS constraints with a sequence separation larger than 11 amino acids. These long-range constraints contain more information than short-range constraints (sequence separation $<$11 amino acids). Short range constraints mainly contain information within or between adjacent α-helices. After we filter CLMS constraints by sequence separation, we obtain 320/107/248 CLMS distance constraints for domains A/B/C for purified HSA and 248/68/163 CLMS constraints for HSA in blood serum. We use CLMS constraints at 20% false discovery rate (FDR) for structure reconstruction experiments.

Please note that false cross-link matches from FDR analysis might or might not exceed the upper 20 Å distance bound. At the same time, a correctly matched cross-link (in terms of FDR) might exceed this distance threshold. This might be caused by conformational flexibility of the cross-linked region that is close in space during cross-linking but further apart in the crystal structure. Thus, the cross-link will still be overlength although the cross-link match is correct. Other unknown effects might also be responsible for overlength cross-links.

Our structure determination algorithm combines information from CLMS constraints, physicochemical information from the energy function, and evolutionary/physicochemical information from EPC-map contacts. As we will see later, leveraging and combining these information sources leads to effective conformational space search. The results in this thesis serve as the first preliminary evidence that the combination of high-density CLMS constraints and computation may result in a fast and inexpensive structure determination method.

Structure Selection of Models from CLMS Data

We tested several structure selection methods to select a single structure and the best out of five structures from model ensembles that are generated with CLMS data. We tested the following methods: Rosetta's physically realistic all-atom energy function, clustering with Durandal [11], Lorentz energy of satisfied CLMS constraints, a knowledge-based potential (ProSA) [193], and an orientation-dependent statistical potential (GOAP) [241].

Our experience with MBS shows that the low-energy ensemble usually contains some low-RMSD structures, even if MBS is unable to rank the lowest-RMSD structure first. Therefore, we also tested two-step structure selection approaches with Rosetta energy as the first step (for example, Rosetta energy+CLMS constraints and Rosetta energy+GOAP).

In this two-step approach, we consider the ten lowest-energy structures by Rosetta energy (course filtering) and re-rank the ten structures with a second methods (for example GOAP).

7.4 Results and Discussion

In this section, we present the results of our structure determination approach using high-density CLMS and conformational space search. We present the results in two parts. We first show the results of cross-linking HSA with sulfo-SDA. We then present our results of reconstructing the structure of HSA domains by the combination of CLMS data and conformational space search.

In section 7.4.1, we introduce the model system of our study, human serum albumin (HSA). In section 7.4.2, we show that sulfo-SDA increases the cross-link density. Section 7.4.3 discusses the impact of incomplete fragmentation of cross-linked peptides, which leads to uncertain site assignment. In section 7.4.4, we show that CLMS data is able to provide evidence for secondary structure determination.

In section 7.4.5, we discuss density of sulfo-SDA cross-links with respect to their utility for structure determination. Section 7.4.6 presents our results of reconstructing the native structure of HSA domains with CLMS data and conformational space search. We demonstrate in section 7.4.7 that the high cross-link density is required to sample near native structures of HSA. Section 7.4.8 analyzes the trade-off between cross-link quantity and accuracy. In section 7.4.9, we analyze and discuss strategies to select representative models from the structure pool that is generated by CLMS calculations.

7.4.1 Human Serum Albumin as a Model System

We choose HSA as a model system because it is (in cross-linking terms) a medium sized protein (66 kDa), it has a crystal structure, and it is also commercially available in purified form. HSA also makes up more than half of the protein content in human blood plasma.

The detection of cross-links is easier for low than for high complex mixtures. Therefore, we can enrich the target protein *after* cross-liking to reduce sample complexity and increase cross-link detection.

We would like to point out that structure determination (X-ray crystallography or NMR) usually requires a high amount of highly purified protein. The most important difference of these methods to the CLMS approach is that purification needs to be performed *before* structure analysis. Thus, the structural integrity of the protein needs to be ensured by the purification procedure. In contrast, CLMS fixes the structure first. Thus, we can use

procedures like SDS-PAGE that disrupts protein structure. HSA is abundant enough in blood serum such that purification by SDS-PAGE is sufficient for CLMS analysis.

7.4.2 Sulfo-SDA Leads to High Cross-Link Density

High-density CLMS results for purified HSA

We argue that the reduced selectivity of the photo-reactive diazirine group in sulfo-SDA significantly increases the number of cross-links (see Figure 7.2). The CLMS analysis with sulfo-SDA produces 205/500/881/1495 at 1/5/10/20% false discovery rate (FDR) for purified HSA (Figure 7.3 and 7.4). In contrast, previous studies on HSA with the highly selective cross-linker Bis(sulfosuccinimidyl)suberate (BS3) only report 43 distance constraints [59]. This comparison demonstrates that photo-reactive diazirine leads to a 10-fold increase in cross-links. In our experiments, we used 87 MS acquisitions, each with approximately 5 μg HSA, after extracting 15 μg HSA from the SDS-gel. We did not optimize our approach to reduce the number of acquisitions. Thus, optimization of this process will be an important goal of future research. We show later that the high number of cross-links at 10/20% FDR is more valuable for structure determination than the cross-links at 1/5% FDR (see section 7.4.8). Therefore, our discussion of the results focuses on CLMS data at these FDR rates (Figure 7.3 and 7.4).

High-Density CLMS Results from Serum HSA Samples

For HSA in blood serum, we identify 644/1304 constraints at 10/20% FDR. For this result, we used 117 MS acquisitions (with 5 μg HSA after extracting 15 μg HSA from the SDS-gel, like in the case of purified HSA). Interestingly, most cross-links from HSA in blood serum agree with those from purified HSA (Figure 7.5). This indicates that the HSA structure can be successfully probed by CLMS in its native environment, human blood serum. We think this is an important advance in the state-of-the-art because established structure determination methods are not able (or to a very limited degree) to obtain structural details of proteins in the complex mixtures that constitute biological environments.

Additional Acquisitions Increase the Number of Unique Cross-Link Pairs

The CLMS results presented in this work are the result of 87 and 117 MS acquisitions for purified and serum HSA samples, respectively. This is a significant number of MS measurements. However, please note that we did not optimize the acquisition protocol yet. The reason for the high number of required acquisitions is that proteomics suffers from

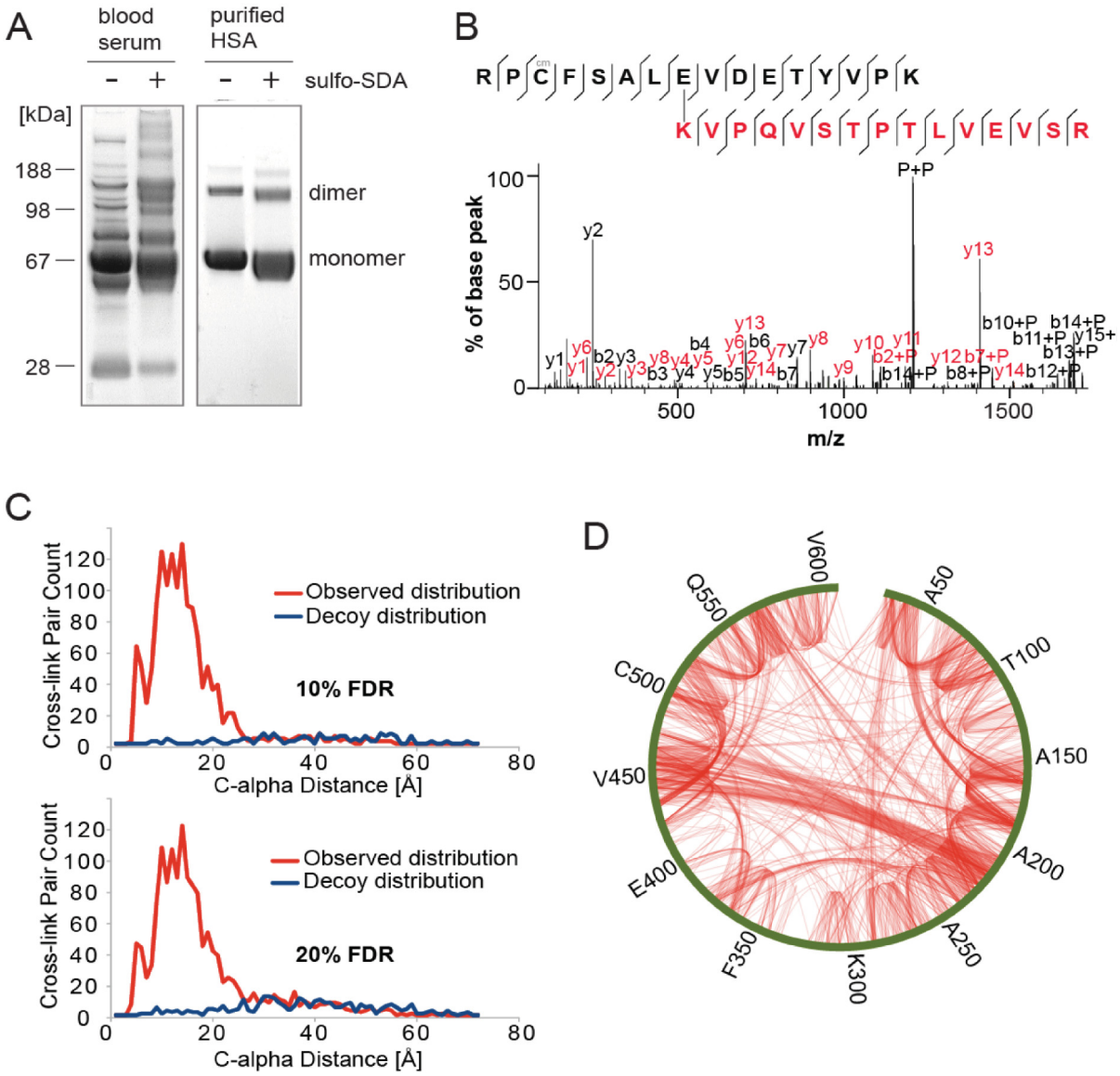

Fig. 7.3 Sulfo-SDA cross-links of purified HSA and HSA in blood serum. **A**: Blood serum proteins and purified HSA, with (+) and without (-) sulfo-SDA cross-linking. **B**: High-resolution fragmentation spectrum of SDA cross-linked peptides that reveals the intramolecular proximity of K437 and E492. **C**: FDR analysis showing observed distance distribution in comparison to the decoy distance distribution. The plot shows the residue-residue C-alpha distances of cross-linked residues and decoy database hits as observed in the crystal structure PDB|1AO6 for the respective residue pairs. **D**: Cross-link network (n= 1,495, 20% FDR) for purified HSA. Green outer line represents the sequence of HSA. Figure adapted from: Belsom et al. [10].

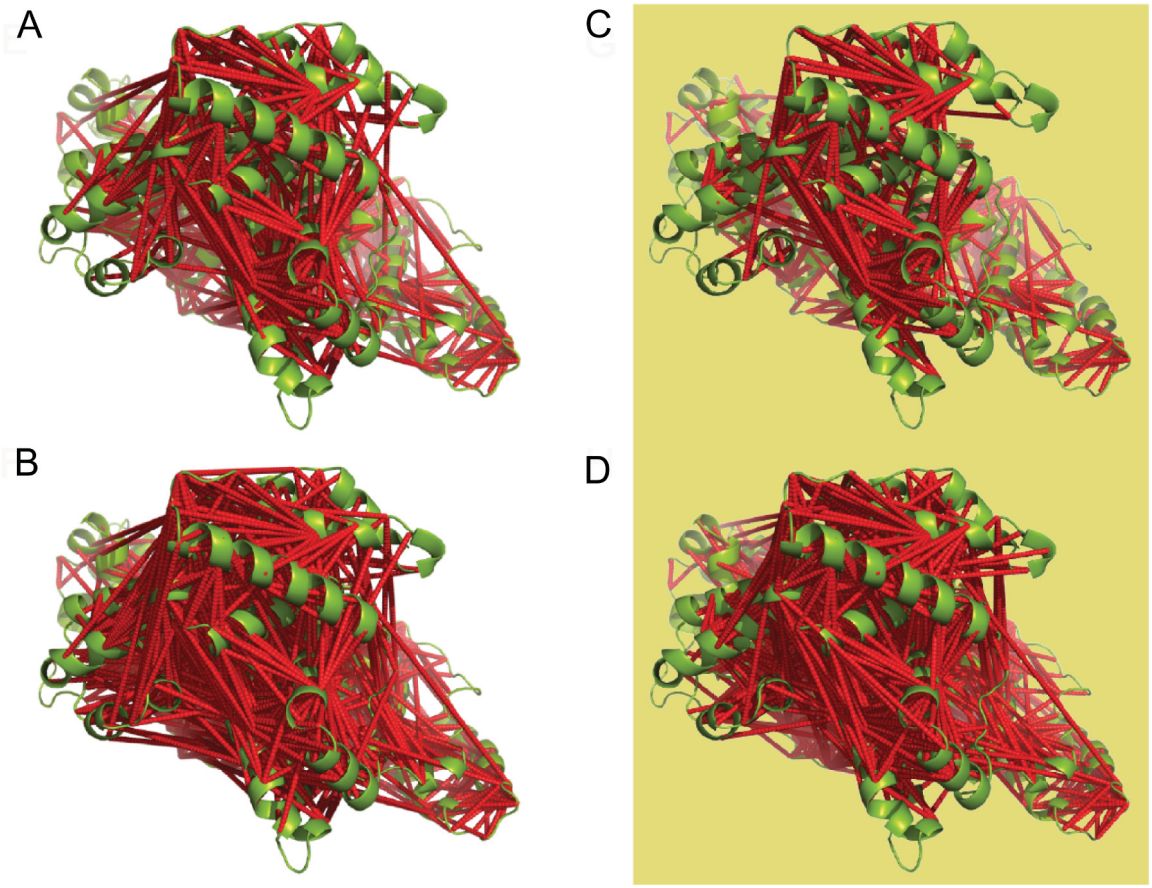

Fig. 7.4 Visualization of sulfo-SDA cross-links on the HSA crystal structure. **A** and **B**: Cross-linked residue pairs of purified HSA in PDB|1AO6: **A**: n= 881, 10% FDR; **B**: n= 1,495, 20% FDR. **C** and **D**: Cross-linked residue pairs of blood serum HSA in PDB|1AO6. **C**: n= 644, 10% FDR; **D**: n= 1,304, 20% FDR. Figure adapted from: Belsom et al. [10].

stochastic detection of analytes. This is especially true if these are only present at low levels. However, each acquisition adds further unique cross-links with half of the cross-links identified after 13.2 +/- 1.6 runs (purified) and 11.5 +/- 2 runs (serum) (Figure 7.6A and B). Note that this is not unique to cross-links. Stochastic acquisition also affects linear peptides albeit to a lesser extent because their intensity is generally higher (Figure 7.6C and D).

7.4.3 Analysis of Uncertainty in Site Assignment

To assign a cross-link between two residues we need to identify the cross-link peptides and the exact linkage site that connects them. Carbenes from photoactivated diazirine can react with any atom in any amino acid. This makes the exact site assignment challenging, which is usually guided by the limited reactivity of selective cross-linking reagents. To unambiguously

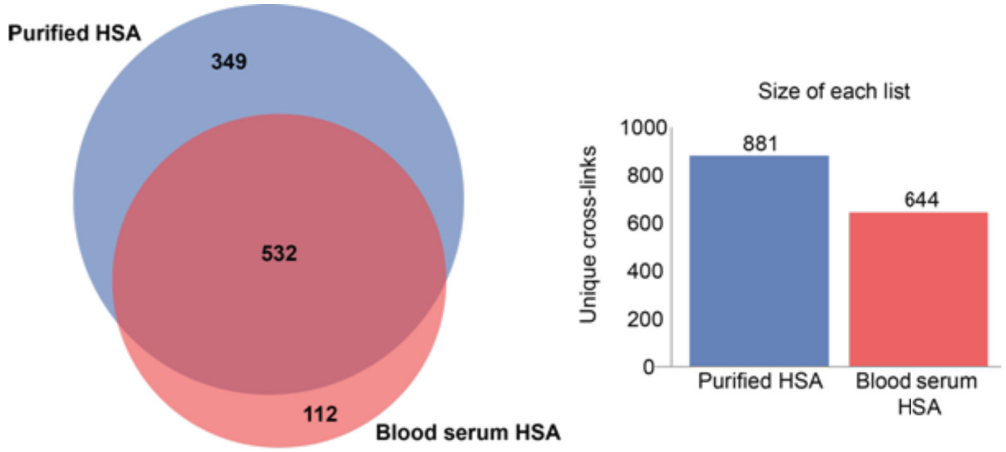

Fig. 7.5 Venn Diagram showing the overlap of CLMS constraints derived from purified HSA and from HSA in blood serum at 10% false discovery rate. Figure source: Belsom et al. [10].

pinpoint the cross-linked amino acids, we need fragmentation spectra on either side of the peptide pair.

We manually investigated 78 randomly selected cross-links and 368 supporting spectra for fragmentation evidence. In 38/78 (49%) of the unique cross-links and 77/368 (21%) of matching spectra, at least one fragmentation event supports the assigned site. If there is no fragmentation evidence, the algorithm looks for flanking fragmentation events and places the linkage site on the first amino acid in this region. This procedure is likely to produce errors in site assignment. Therefore, we here study the impact of imprecise site assignment on the resulting distance distribution of cross-linked residue pairs.

We studied imprecise site assignment by randomly shifting the diazirine sites by 3-39 residues and assessed the impact on the distance distribution. We used all cross-links with more than 12 residues sequence separation at 20% FDR for this experiment (Figure 7.7). If we compare the distance distribution of our original site assignments and randomly shifted assignments, we find that the distributions differ significantly (at $p = 0.05$) if we consider a shift window of 11 residues. The median shift of the distance distribution at this window size is 1.00 +/- 0.16 Å. Since the median length of the cross-linked peptides is 12 residues, even a random site assignment without supporting fragmentation spectra results would not impact the overall distance distribution.

As we will see later, our results indicate that our CLMS data with uncertain site assignments is still useful for structure modeling. This indicates that the explicit treatment of

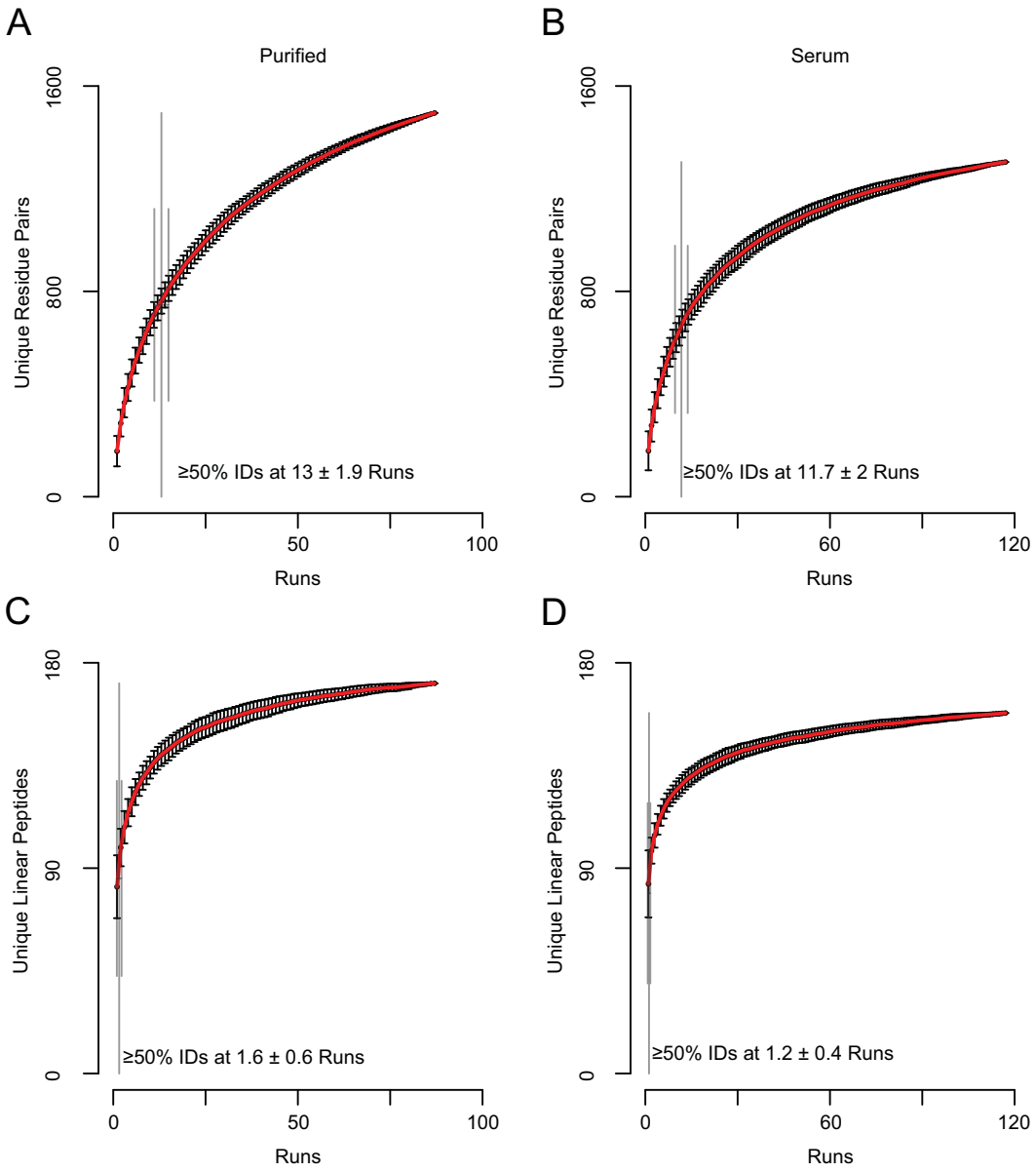

Fig. 7.6 Residue pair and linear peptide identifications accumulated over runs. **A** and **B**: Total number of unique residue pairs (20% FDR) increases with each successive LC-MS run for cross-linked purified HSA (**A**) and cross-linked blood serum HSA (**B**). **D** and **C**: Total number of linear peptides identified (5% FDR) in the same raw data as used in A and B. The number of linear peptides increases with each successive LC-MS run (**C**: purified HSA, **D**: blood serum HSA run). The order of LC-MS runs in the series was permuted 100 times and the mean increase per run in all permutations is plotted. The standard deviation for each point is plotted as error bars. Four missed cleavages were allowed and only unique residue pairs or peptide sequences, respectively, were counted (i.e. modifications were ignored during the counting of unique IDs). This allowed for linear peptides the maximum possible number of peptides (sequence of HSA, requiring at least 7 residues, allowing for up to 4 missed cleavages) to be predicted at 335. This means that still not all theoretically possible peptides were seen, keeping in mind that trypsin missing more than one cleavage site is a very rare event and thus such peptides are seen with very low intensities. Acquiring additional runs creates more opportunity to identify peptides that are observed with signals near the detection limit. Figure source: Belsom et al. [10].

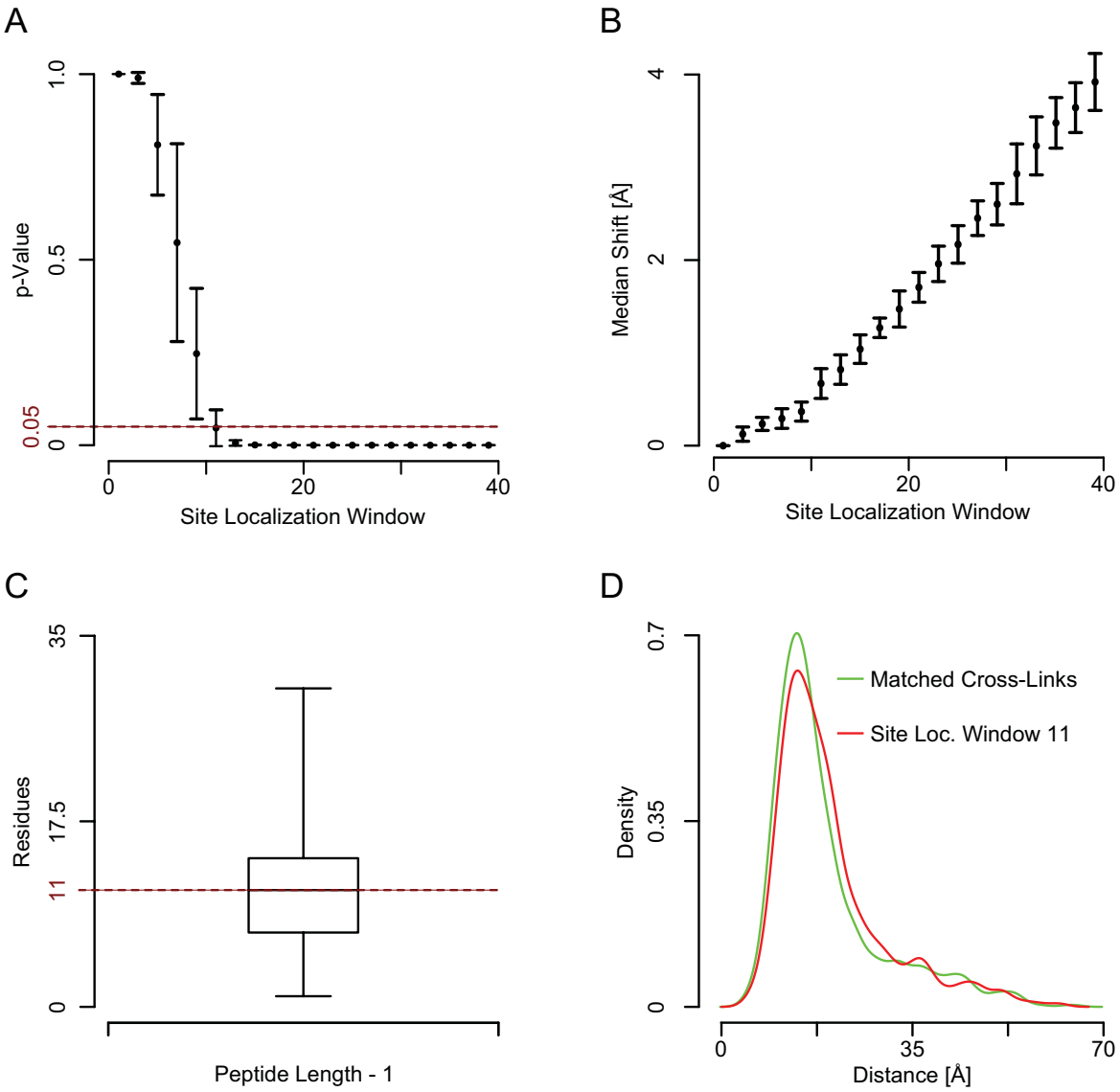

Fig. 7.7 Impact of Link Site Ambiguity. **A**: p-values for Kolmogorov-Smirnov test comparing the distance distribution of the identified residue pairs (cross-links, 20% FDR) as measured in PDB|1AO6 (α-carbon distances) with distance distributions where for each unique residue pair, one site (the diazirine-linked site) was randomly reassigned to a residue within a window centered on the original match site. For each window size, 100 distributions of locally randomized residue pairs were calculated. The median p-value is plotted for each window size along with the 1.4826*MAD as error bars. **B**: The difference between the median of the original residue pair distances and the median of the randomized site distribution for each window-size. **C**: Box plot for the length - 1 of the diazirine linked peptides for all PSMs. Length -1 is used, as a link to the carboxy-terminal residue would inhibit trypsin cleavage and is therefore excluded as a possible linkage site. The median number of linkable residues is 11. **D**: Density plot of the distances of the identified residue pairs overlayed with one example of the 100 randomized distributions for a link site tolerance of 11 residues. Figure source: Belsom et al. [10].

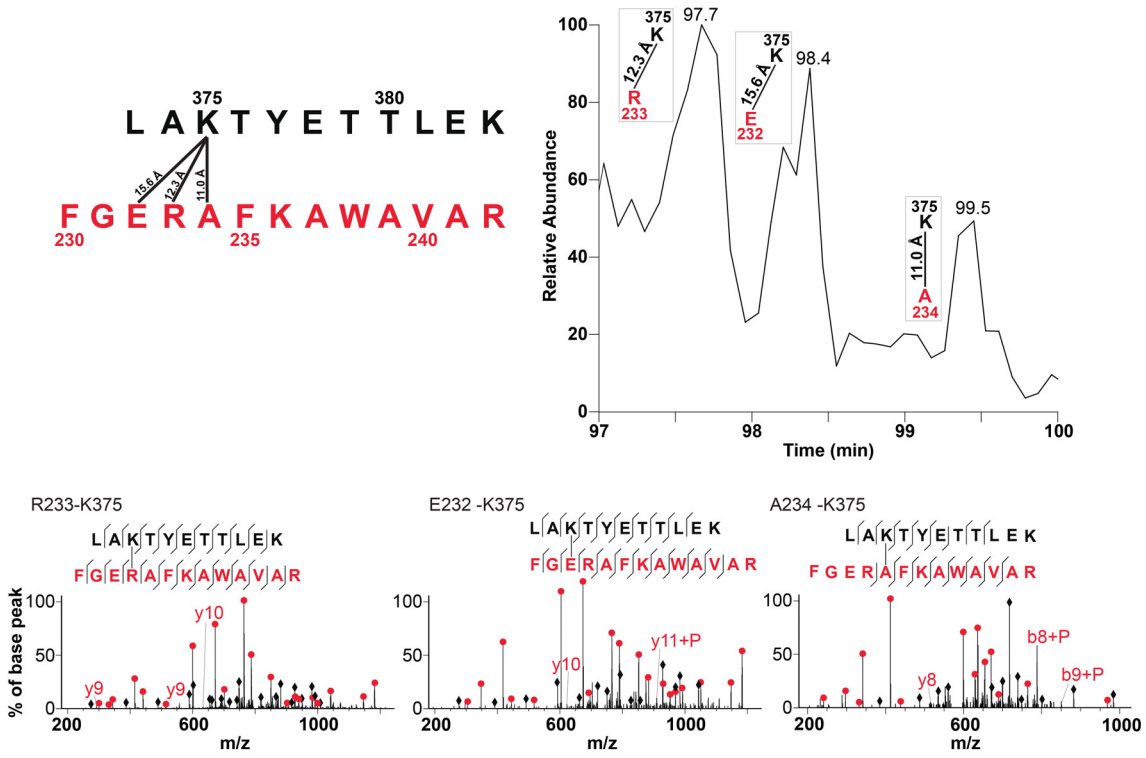

Fig. 7.8 Extracted ion chromatogram of a cross-linked peptide pair. Cross-linked peptide pair labeled with sequence number with first peptide match shown in black and second peptide match in red. We found K375 cross-linked to three residues (R233, E232 and A234) in the second peptide in a single LC-MS run. Peaks in the extracted ion chromatogram (97.67, 98.38 and 99.45 mins) are labeled with the sites of cross-linking in the peptide pair matched by the database search software, Xi. C_α distances are indicated on cross-linked residue pairs. Fragmentation spectra for each cross-linked peptide pair are shown at the bottom as evidence of identification. Figure source: Belsom et al. [10].

uncertain cross-links by the Lorentzian energy function is able to compensate for such errors. Nevertheless, we cannot rule out that errors in site assignment do not impact modeling. Even if the overall distribution does not change, the distance constraint of individual residue pairs might change significantly by uncertain site assignment. This indeed could affect structure modeling. The effect of uncertain site assignment on modeling should be systematically tested in future research.

However, we also find that manual inspection reveals linkage sites in our data. We looked at identical peptide pairs with different linkage sites that we separated by liquid chromatography (Figure 7.8). In one peptide pair, we found cross-links between K375 on the first peptide and three residues (E232, R233 and A234) on the second peptide. We identified these cross-links in a single LC-MS run. We matched the cross-linked peptide spectrum scan numbers to the raw file and revealed three distinct peaks on the LC-MS chromatogram.

Each of these peaks corresponds to a different cross-link position. This shows that adjacent cross-link sites can be identified by LC-MS analysis.

7.4.4 Evidence of Secondary Structure by CLMS

We also investigated whether the increased cross-link density of sulfo-SDA allows for direct structural analysis. We manually validated the placement of diazirine reactive sites and found evidence of the underlying secondary structure (Figure 7.9). Note that this is a single instance as a result of manual inspection. We found the lysine (K186) on one peptide cross-linked to five other amino acids on the other peptide: F151, E155, E156, L159 and Y162. Assembling the amino acid sequence of the second peptide on a α-helical wheel, beginning with A150 and ending with Y162, explains this pattern. Thus, this pattern is indicative for the cross-linking of K186 to an α-helix in close proximity. Examination of the X-ray crystal structure of HSA confirms this finding.

7.4.5 Using CLMS and Conformational Space Search to Determine HSA Domain Structures

We showed that sulfo-SDA dramatically increases the number of cross-links. Given the size of HSA (576 residues) and number of cross-links that we identify, the sulfo-SDA CLMS approach produces on average 1.5/2.5 constraints per residue at 10/20% FDR. This approaches the number of constraints obtained in NMR experiments (3-20 constraints per residue) [111]. However, HSA has a molecular weight of 66 kDa and is therefore much larger than proteins that NMR typically investigates. The number of CLMS constraints is still insufficient to determine protein structure with standard NMR protocols (data not shown).

We now demonstrate that CLMS data from our experiment does contain sufficient information to reconstruct protein structure. To unlock this information, we combine CLMS data with other information sources, such as fragments, contacts, and the physicochemistry captured in energy potentials of state-of-the-art *ab initio* structure prediction. We use CLMS data as input for constraint-guided model-based search (guided-MBS) (see Figure 7.1, and section 6.2.1). We would like to remind the reader that earlier studies used cross-link data to verify models from homology modeling, which requires homologous template structure information [191, 231]. However, in both studies CLMS information did not affect the building of structure models. In our approach, CLMS information is the key source to build and verify protein structure with *ab initio* structure prediction. In contrast to homology modeling, this process is applicable with proteins without structural homologs.

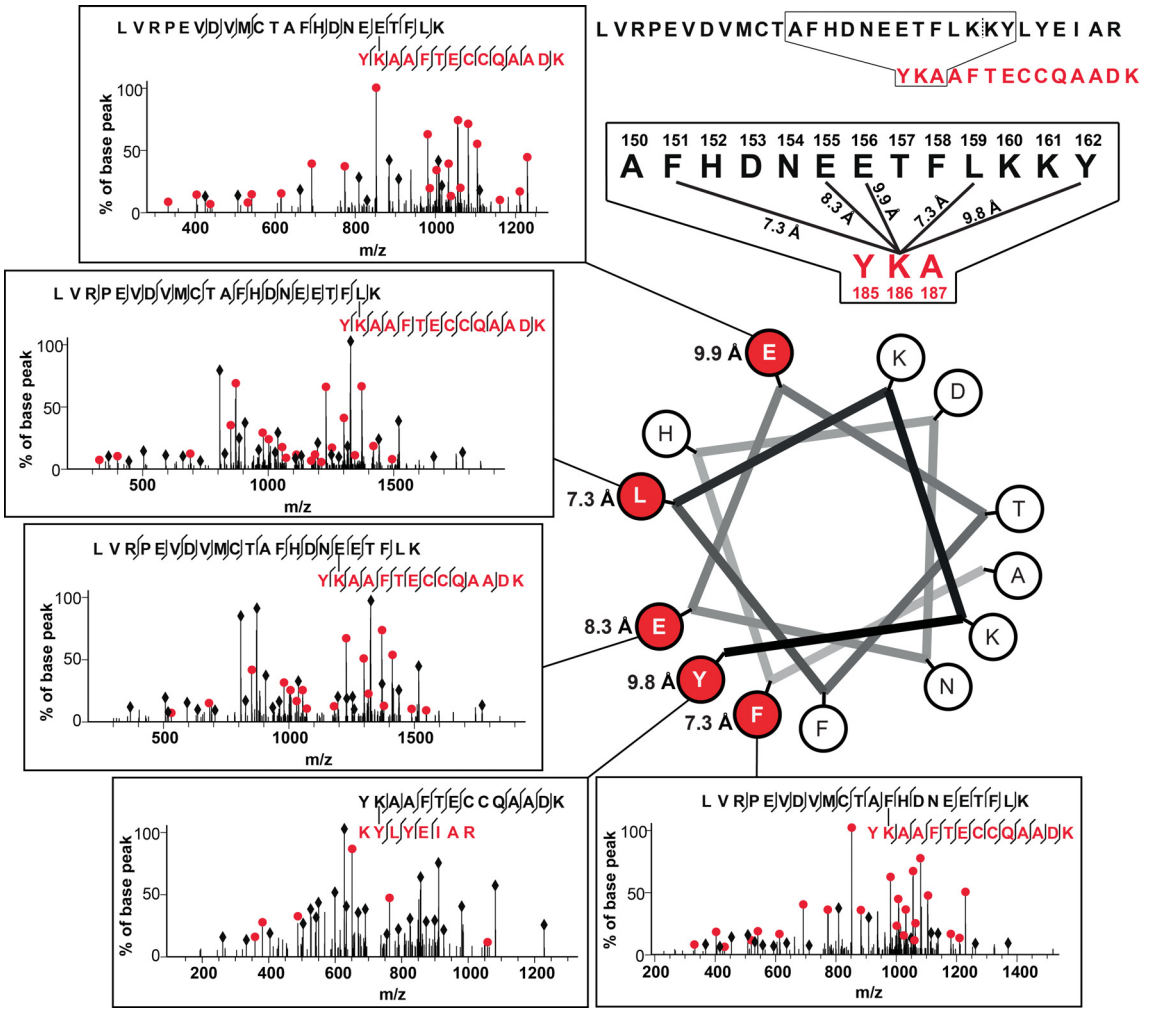

Fig. 7.9 Identified cross-linked sites suggest α-helical secondary structure. K186 from peptide two (sequence shown in red) found cross-linked to five residues (F151, E155, E156, L159 and Y162) from peptide one (sequence shown in black). The full sequences of peptides identified as cross-linked peptide pairs are shown at the top right, with an expansion of the sequence showing residues cross-linked to K186 shown underneath. Residues from the sequence of peptide one are shown assembled in a representation of an α-helix, beginning with A150 and ending with Y162. Residues identified as cross-linked to K186 are colored red, with the associated fragmentation spectra for each cross-linked peptide pair shown as evidence of identification. C_α distances are associated with each cross-linked residue pair. Figure source: Belsom et al. [10].

Ab initio structure prediction is typically only possible for small proteins [223], because the search space is extremely large and the energy landscape is rugged. However, we hypothesize that CLMS information directs search towards near-native conformations. We leverage CLMS information as distance constraints in a Lorentzian function (see section 5.3.8). The distance constraints deepen wells in the low-resolution energy landscape that correspond to structures that satisfy the constraints, which steers search towards biologically relevant regions.

At the same time, guided-MBS is designed to deal with noise in input data (CLMS and contacts). The Lorentzian function maximizes the number of satisfied constraints and does not penalize constraints that are not satisfied. In addition, guided MBS only relies on Rosetta's all-atom energy function to assess the quality of a funnel and refine low-resolution decoys to high-resolution structures.

7.4.6 Determination of HSA Domain Structure with CLMS and Search

We use guided-MBS to reconstruct the structures of the three domains of HSA using CLMS constraints at 20% FDR. We use CLMS data derived from experiments on purified HSA and from HSA samples in human blood serum. We divide HSA because the full-length protein is too large for existing computational methods. For purified HSA, our methods identifies 320 CLMS distance constraints for domain A (22.6 kDa, 197 residues), 107 constraints for domain B (21.5 kDa, 189 residues), and 248 constraints for domain C (21.7 kDa, 192 residues). Note that we only use CLMS constraints with a sequence separation of > 11 residues.

Table 7.2 Low-energy ensembles of domains A/B/C of HSA. Table source: Belsom et al. [10].

Domain	Residues[a]	Used residues for full scoring/RMSD[b]	Purified CLMS[c]	Serum CLMS[c]	No CLMS[c]	BS3[c]
A	1-197	2-71:115-194	2.5/3.9/7.7	3.4/4.1/10.2	7.9/12.5/16.6	10.2/12.1/ 16.4
B	198-386	200-262:308-381	4.9/5.7/8.4	5.2/8.3/11.5	7.4/11.8/14.2	11.9/15.7/16.5
C	387-578	389-458:508-571	2.9/3.2/5.7	3.8/5.0/8.0	15.5/16.3/16.9	10.4/11.1/17.4

[a] In our notation, the first residue in the HSA (PDB|1AO6) crystal structure is denoted as residue 1.

[b] These residues are used for full scoring in the all-atom phase and for RMSD calculation. From the non-listed residues, only repulsive terms of the energy function are used.

[c] The numbers refer to the min/median/max RMSD values in the ensemble of the ten lowest-energy structures.

Refer to Figure 7.10 for the rest of this section. CLMS data increases the sampling of low-RMSD structures, compared to conformational space search without constraints (Figure 7.10A-C). The computed structures for domains A/B/C have an RMSD of 2.8/5.6/2.9 Å

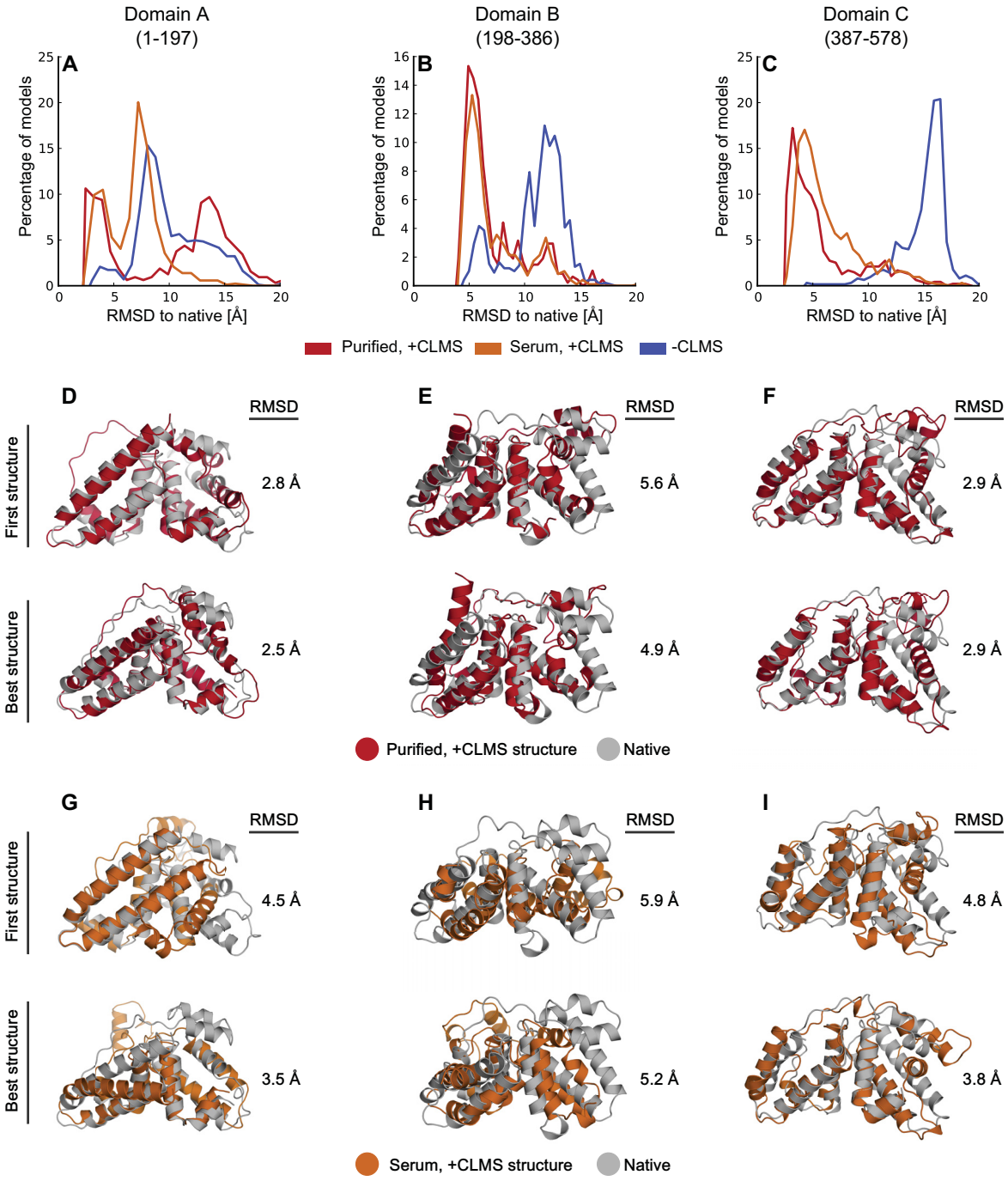

Fig. 7.10 Determined structures for the individual domains of HSA by using cross-link constraints and conformational space search. **A-C**: Deviation of domain structures obtained with our novel procedure to the crystal structure of HSA (PDB|1AO6). CLMS data from purified and serum HSA (red and orange curve) increases the sampling of low-RMSD structures, compared to structures obtained without CLMS data (blue curve). **D-F**: First and best determined structures calculated with CLMS data from purified HSA, aligned to the crystal structures of the HSA domains. For each domain, "First structure" refers to the single structure we selected using Rosetta energy+CLMS constraints. "Best structure" refers to the lowest RMSD structure to PDB|1AO6 among the best five structures ranked by Rosetta energy+GOAP (see section 7.4.9 for method details). **G-I**: First and best determined structures calculated with CLMS data from HSA samples in blood serum. Large loops in the crystal structure and terminal residues predicted to be disordered are removed for RMSD computation. The following residues are used for calculation of the RMSD: 2-71:115-194 for domain A, 200-262:308-381 for domain B and 389-458:508-571 for domain C. Figure source: Belsom et al. [10].

to the native structure (PDB|1AO6, see Figure 7.10D-F, Table 7.2). Most deviations to the crystal structure are in the long loop regions, probably because of the inherent flexibility of loops. We also find deviations from the crystal structure in the interface regions between domains. This is expected by some degree, because we removed the domains from their context, which might be required to correctly pack the domain interface. Nevertheless, the low energy ensembles converge and displays sampling around the native structure (Figure 7.11). Domain B has a higher RMSD than the other domains (5.6 Å as opposed to 2.8/2.9 Å), which is presumably a result of the relatively small number CLMS constraints for this domain (107 CLMS constraints).

Another important feature of the CLMS approach is that, once cross-links are formed, we can isolate the protein under denaturing conditions because the structural information is already stored in the cross-links. Thus, we can cross-link HSA in serum and enrich it using SDS-PAGE (Figure 7.1). We repeated the structure determination experiments with CLMS data from serum samples to test whether this set of CLMS constraints also has the ability to determine the HSA domain structure. Therefore, we are able to collect structural information of a protein in its native, biological environment. The CLMS in-serum ensemble and the purified CLMS ensemble show significant overlap in their RMSD distribution. This demonstrates that CLMS information from serum samples contains information for protein structure determination. The RMSD to the native structure is 4.5 and 4.8 Å, for domains A and C, respectively (Figure 7.10G-I). As for CLMS data from purified samples, the RMSD of domain B is higher (5.9 Å). This is probably caused by the smaller number of available CLMS constraints (248/68/163 constraints for domains A/B/C). The agreement with the native structure is higher for the best-of-five structures (RMSD 3.5/5.2/3.8 Å for domains A/B/C, Figure 7.10G-I).

7.4.7 High CLMS Density Is Required for Sampling of near Native Structures

We demonstrated that CLMS constraints from sulfo-SDA are able to reconstruct the structure HSA domains. However, the information in standard BS3 cross-linker could also be sufficient to reconstruct the native structure. In that case, there would be limited value from high-density CLMS data for structure determination. Thus, we also tested the ability to determine the structure of HSA domains without CLMS constraints, and also with BS3 CLMS constraints.

If we do not use the additional information of CLMS constraints, the best models from conformational space search have low backbone accuracy (RMSD 7.9/7.4/15.5 Å for the best structure among the ten lowest-energy structures, see Figure 7.10 and Table 7.2).

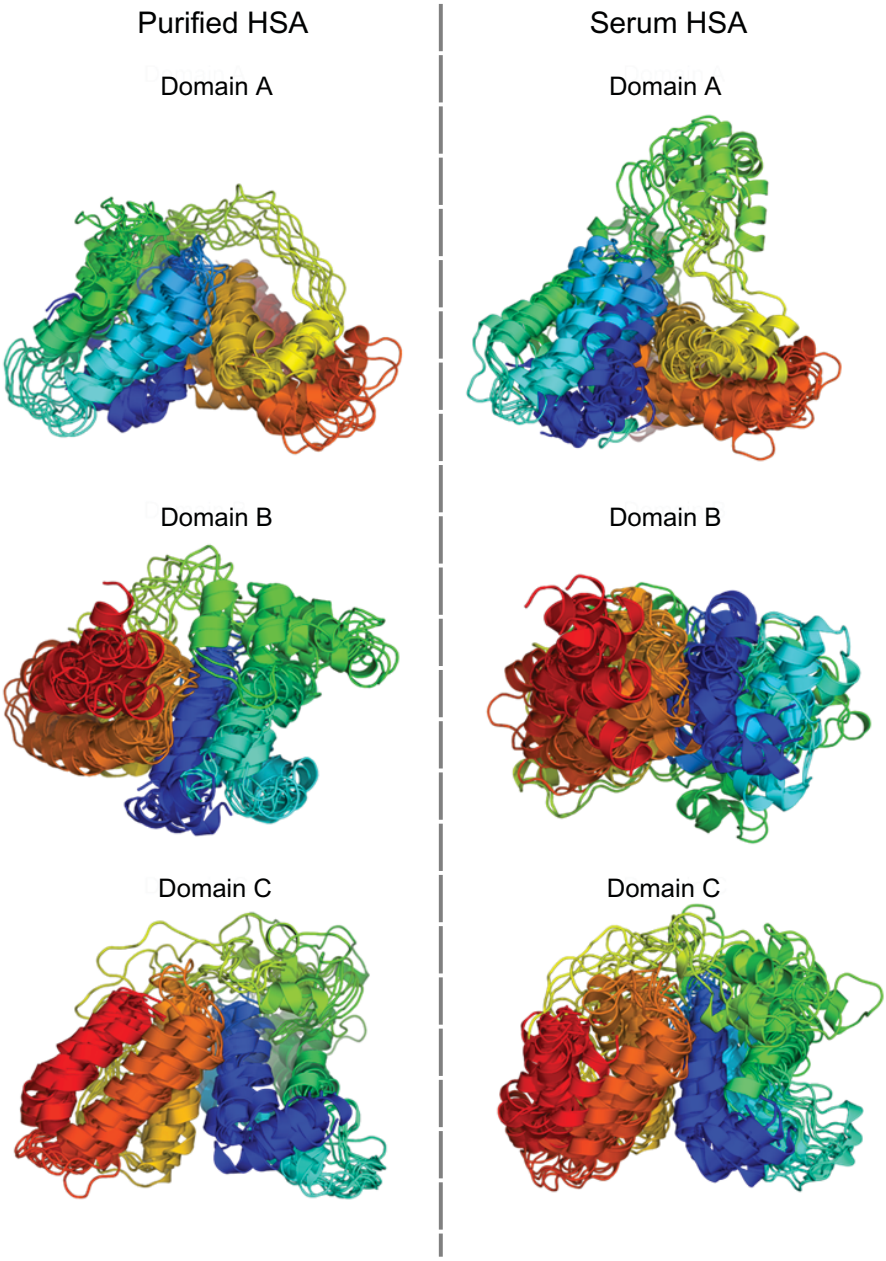

Fig. 7.11 Low-energy ensembles of the domains of HSA. Low-energy ensembles were computed with CLMS data from purified HSA samples (left column) and from HSA in serum (right column). For each domain, we show the 10 lowest-energy structures from our calculations. Structures in low-energy energy ensembles from purified HSA CLMS data are in good agreement with each other, indicating convergence of the algorithm. Data from serum HSA is slightly noisier and computations produce more heterogeneous ensembles then for purified HSA. Nevertheless, the resulting ensembles show good sampling around the native state. Figure source: Belsom et al. [10].

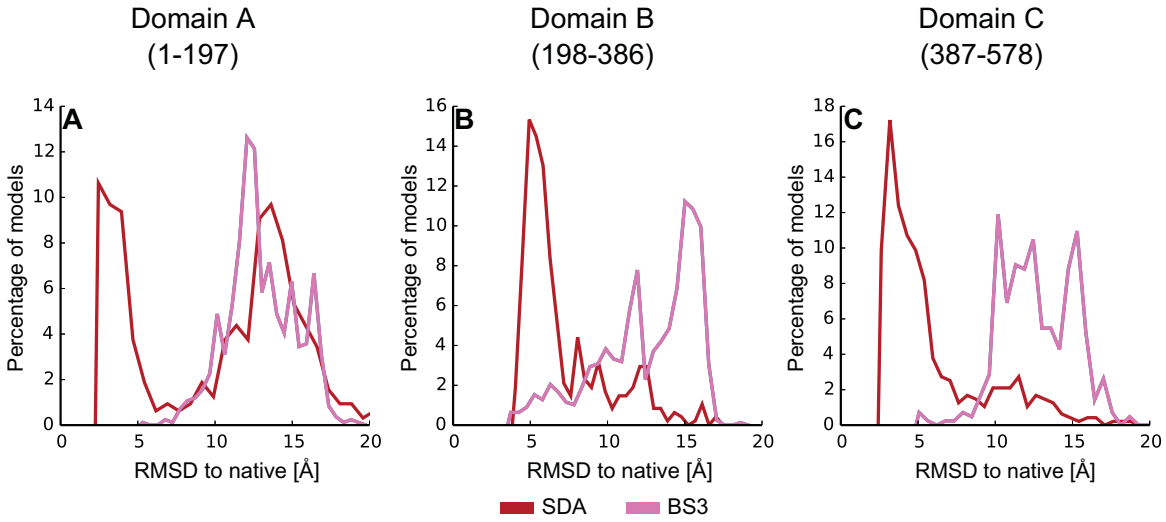

Fig. 7.12 Backbone quality of the structure ensemble generated with SDA cross-linker (our method) and BS3 cross-linker. We obtained both CLMS datasets from purified HSA samples. CLMS data from SDA cross-linking experiments contain an order of magnitude more cross-links (SDA: 320/107/248; BS3 16/14/11 cross-links for domains A/B/C). This leads to an increased sampling of low-RMSD HSA domain structures. Figure source: Belsom et al. [10].

When we use BS3 CLMS data to guide conformational space search, the best structures are comparable in backbone accuracy to structures from unguided calculations (RMSD 10.2/11.9/10.4 Å for the best structure among the ten lowest-energy structures, see Figure 7.12 and Table 7.2). Therefore, the information of BS3 CLMS constraints is not sufficient for *ab initio* structure determination of HSA domains.

7.4.8 Trade-Off Between Cross-Link Quantity and Accuracy

As mentioned in section 7.3.1 and 7.4.6, we use the CLMS data at 20% FDR. Our initial experiments showed that the noise in the CLMS data (at least 20% of the cross-links are miss-matches) does not hinder successful modeling of HSA. However, we systematically investigated the tradeoff between cross-link accuracy and quantity to test the impact of the FDR on backbone quality of the structure ensemble. Understanding this tradeoff is important for future development of the CLMS technology. Thus, we addressed the following question: Is it important to identify fewer but more precise cross-links or should we focus on further increasing cross-link density at the expense of accuracy?

We also tested the impact of the FDR on the backbone quality of the structure ensemble to investigate the tradeoff between cross-link accuracy and quantity. We repeated the MBS calculations with CLMS data at 1/5/10/20% FDR. We then quantified the quality of the ensemble by the RMSD at the 1% percentile (Figure 7.13). This assessment tests the ability

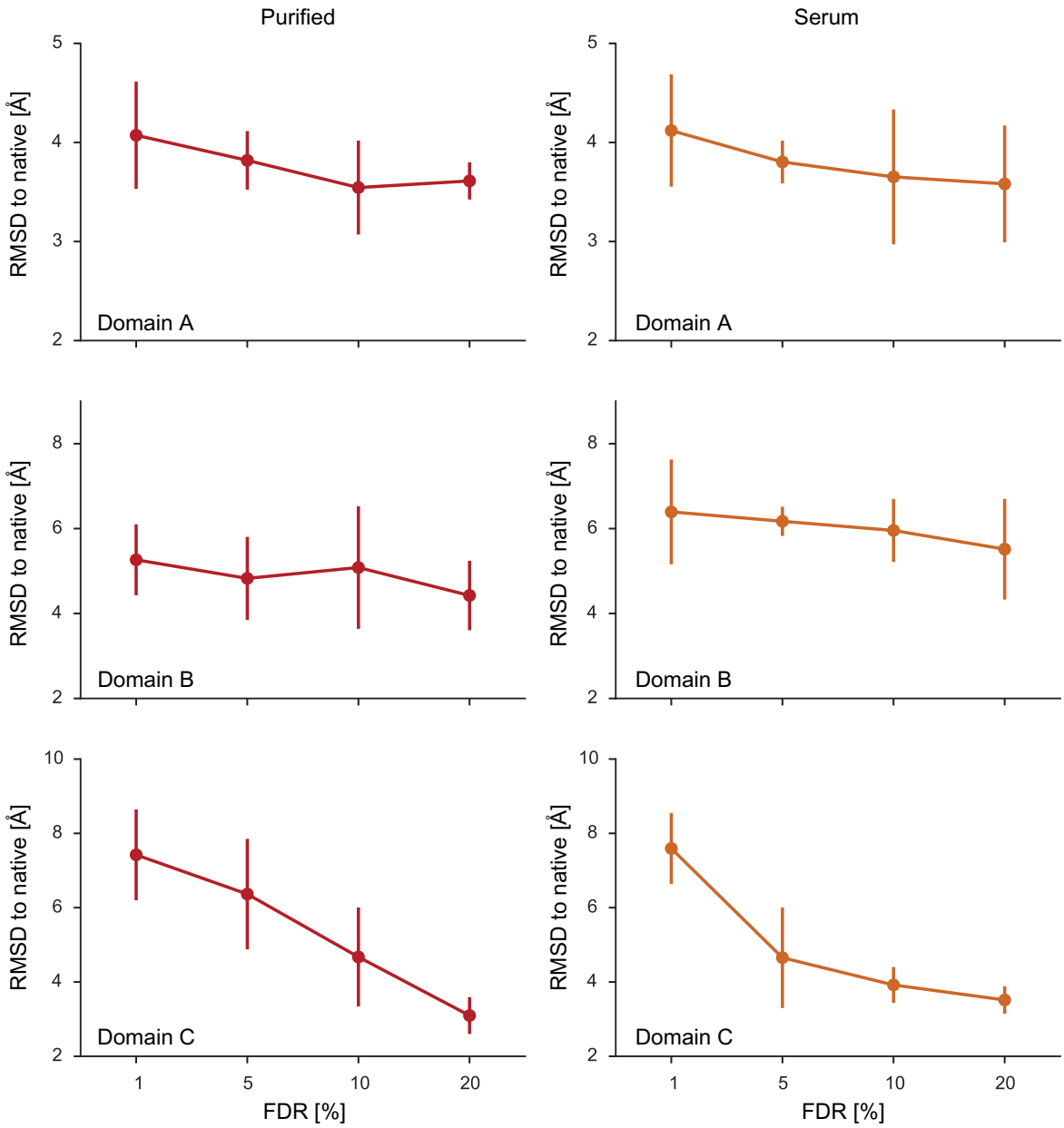

Fig. 7.13 Impact of the FDR on the backbone quality of the sampled structure ensemble. We repeated the modeling experiments with CLMS data at 1/5/10/20 FDR five times (see section 7.2.3). We measured the impact on ensemble quality by the mean and standard deviation of the RMSD to native at the 1% percentile over five runs. This metric assesses the quality of the structure ensemble by the upper RMSD bound of a 1% of generated structures. Generally, CLMS data at higher FDR values improve the quality of the structure ensemble. With the exception of domain A with CLMS data from purified samples, data at 20% FDR results in the lowest-RMSD ensembles. Note that we reduced the number of samples to 2000 samples per MBS stage (5000 for production runs) for this experiment. This is necessary because of the high computational cost of this experiment. Figure source: Belsom et al. [10].

of the algorithm to generate low-RMSD decoys without considering the issue of model selection. We quantify the ensemble quality by the 1% percentile RMSD to reduce the impact of outliers and thus make the quantitative assessment more robust.

Guided-MBS finds the lowest RMSD structures at 10 or 20% FDR. CLMS data at 20% FDR results in the lowest RMSD ensembles (with the exception of domain A from purified samples). The RMSD at the 1% percentile drops from 4.1/5.3/6.4 Å (1% FDR) to 3.6/4.2/3.1 Å (20% FDR), for domains A/B/C. The results indicate that a high number of CLMS constraints (10-20% FDR) is more effective than few, accurate links (1-5%) FDR protein modeling HSA domains. For domain C, this effect is most pronounced. The sampled structures improve dramatically from 6.4 Å at 1% to 3.1 Å at 20% FDR.

Overall, our results suggest that a high number of cross-link constraints, even at significant degrees of noise, is more useful for structure modeling than few, accurate links. The Lorentzian function and noise-tolerant implementation of guided-MBS is presumably contributing the robustness of the structure prediction algorithm, at least partially. Thus, an attractive route to advance CLMS based structure determination would be to develop algorithms that are able to cope with even higher degrees of noise. Those algorithms could benefit from the increased number of cross-links at even higher FDRs.

7.4.9 Selecting Structural Models from CLMS Structure Calculations

We also tested different approaches to select models out of the structure ensemble (see section 7.4.9 and Figure 7.14). For HSA domains, a two-step selection procedure using Rosetta energy and CLMS constraints consistently selects structures with correct topology (RMSD smaller than 6 Å). Structures selected by this method are denoted as "First structure" in Figure 7.10. For purified HSA, this procedure selects structures with good agreement to the native structure for (RMSD of 2.8/5.6/2.9 Å for domains A/B/C). The first structures selected for serum CLMS have lower quality (RMSD of 4.5/5.9/4.8 Å for domains A/B/C).

However, inaccuracies in the energy function and noise in CLMS data might not rank the most native-like structure first. Therefore, we also tested structure selection methods for their ability to rank low-RMSD structures within a low enough number of structures that could be manually inspected. In this work, we tested the ability of structure selection methods to rank the best five structures (Figure 7.15). For selecting the best out of five structures for purified CLMS, Rosetta energy in combination with an orientation-dependent all-atom statistical potential (GOAP) performs best (2.5/4.9/2.9 Å for domains A/B/C). The best out of five structures for serum CLMS are in good agreement with the native structure (RMSD 3.5/5.2/3.8 Å for domains A/B/C).

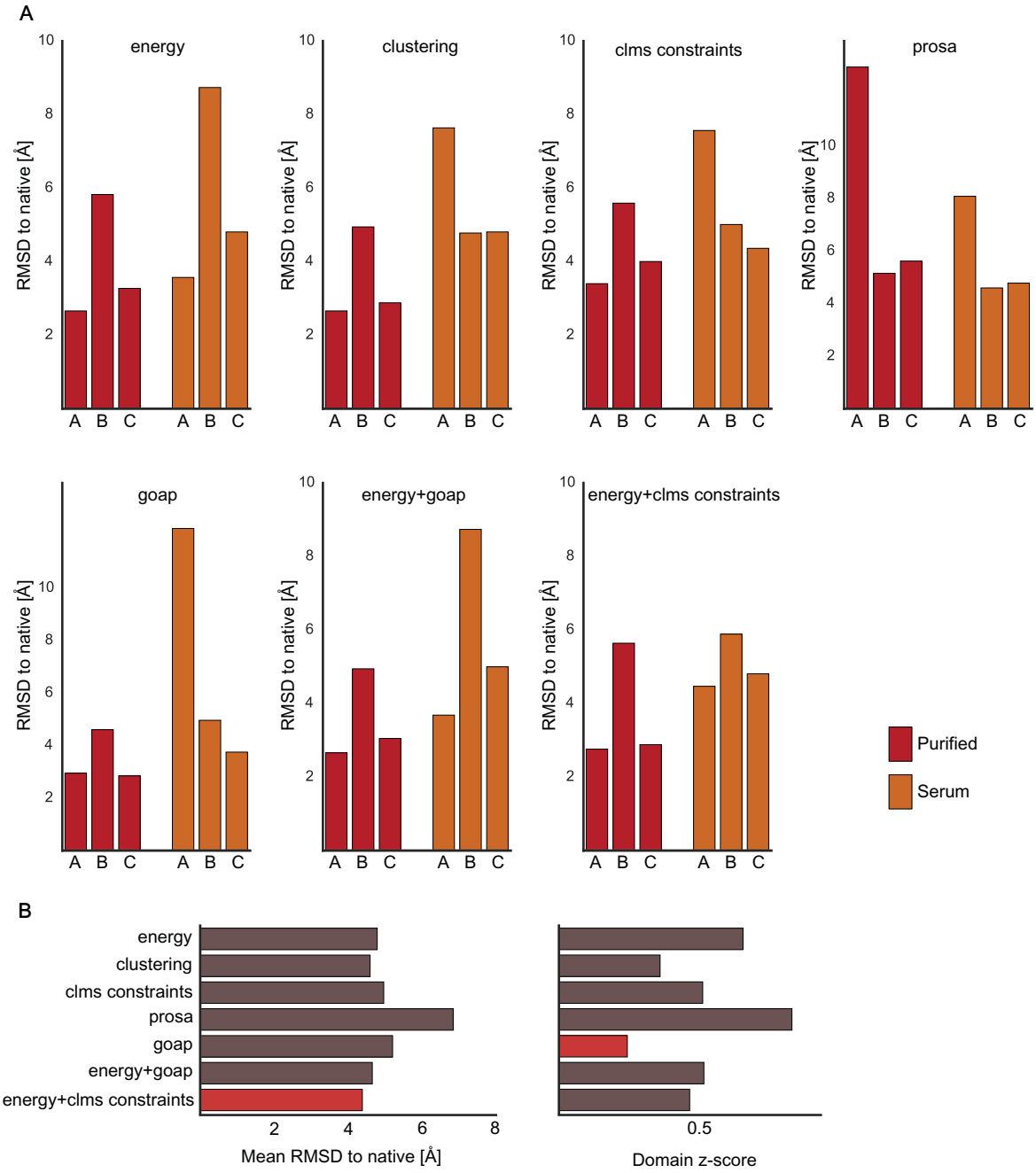

Fig. 7.14 RMSD of the first CLMS structure selected with several structure selection methods. **A**: Individual structure selection results for domains A/B/C with CLMS data from purified (red) and serum (orange) samples (lower is better). **B**: Overall performance (mean RMSD over all domains and domain-wise z-score) of all tested structure selection methods. The best method is shown in red. Overall, Rosetta energy+CLMS constraints selects structures with lowest mean RMSD. GOAP finds lower RMSD structures than energy+CLMS constraints for most domains, but selects a wrong fold for domain A in serum. Overall, energy+CLMS constraints does not necessarily select the best structure, but consistently selects structures with native topology (RMSD smaller than 6 Å). Therefore, energy+CLMS constraints is the most robust method we tested and is used to select the first representative structures in this work. Figure source: Belsom et al. [10].

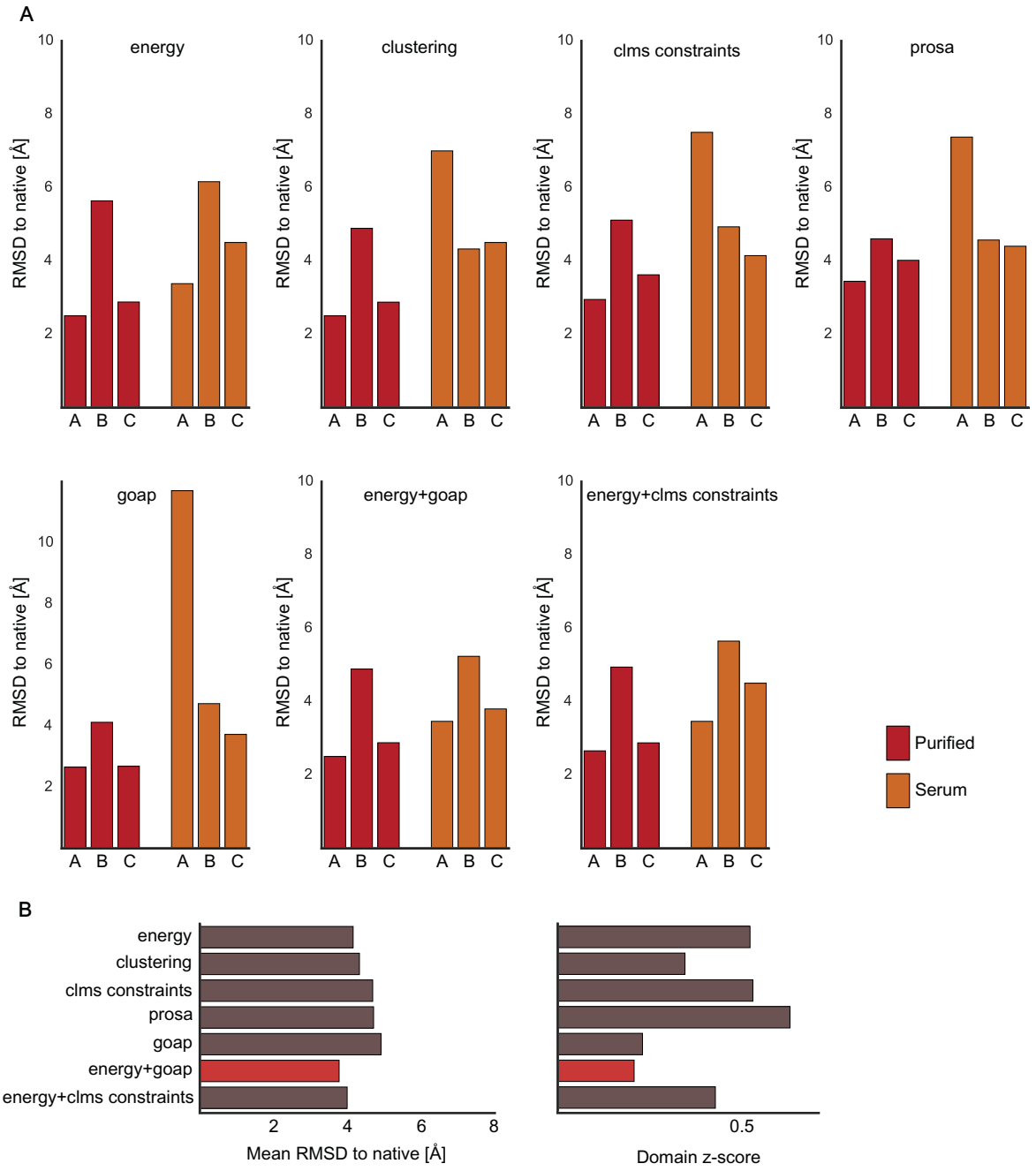

Fig. 7.15 RMSD of the best out of five structure selected with several structure selection methods. **A**: Individual structure selection results for domains A/B/C with CLMS data from purified (red) and serum (orange) samples (lower is better). **B**: Overall performance (mean RMSD over all domains and domain-wise z-score) of all tested decoy selection methods. The best method is shown in red. Overall, Rosetta energy+GOAP is the best method for selecting the best out of five structures. Thus, this method is able to select a small number of structures that can be further evaluated with additional experimental data and/or biological knowledge. Figure source: Belsom et al. [10].

We make the interesting observation that structure selection with CLMS data from purified HSA leads to slightly better results for HSA from serum samples (RMSD improvement from 4.5/5.9/4.8 to 3.6/5.9/4.5 Å). This indicates that CLMS data is valuable for both, conformational space search and model selection, even though we would not be able to proceed this way when probing protein structure in complex environments.

We would like to stress that these model selection results need to be interpreted with caution. Obviously, since we only have high-density CLMS data for one protein (HSA), we can only test model selection on this protein. Thus, the model selection results presented here might need revision when sulfo-SDA CLMS data for more proteins is available. However, other groups also reported the value of CLMS data for the selection of *ab initio* predicted structures [80, 91], which supports our finding. As we have shown earlier, input constraints influence the effectiveness of structure selection methods (see chapter 5, section 6.4.4). Thus, CLMS data on more proteins is to increase the informative value on model selection experiments from CLMS guided calculations.

7.5 Conclusion

We presented a novel hybrid method that combines high-density photo-cross-linking data with conformational space search to determine protein structure. We recapitulated the domain structure of human serum albumin (HSA) in solution to demonstrate this technology. Importantly, the high-density of cross-links of the photo-CLMS approach leads to sufficient information to determine the structure. In contrast, this was not possible using standard homobifunctional cross-linking reagents (BS3).

Perhaps even more importantly, we showed that this approach also allows the study of HSA domains in human blood serum, which constitutes a complex mixture of proteins. This is strong evidence that our method is able to probe proteins in their native environment. This is a significant advantage over other structure determination methods, such as NMR and X-ray, that are —with few exceptions— only able to probe protein structure in artificial *in vitro* environments or protein crystals.

However, our success with HSA merely represents a proof-of-concept and the wide applicability now needs to be demonstrated by probing the structure of other proteins. This might need further improvements of the method. In human blood serum, HSA is the most abundant protein which simplified the enrichment of the protein after cross-linking. We assume that less abundant proteins require a more sophisticated enrichment procedure. There are multiple dimensions along this approach could be optimized, such as sample preparation,

cross-link chemistry, mass spectrometric acquisition, analysis of mass spectrometric data, and use of CLMS constraints during protein structure prediction.

Ultimately, we envision that high-density CLMS and conformational space search hybrid methods will be an effective complement to established structure determination methods. If this technology builds on its current promise, it might become a key tool in structural biology. Being able to probe protein structures in their native environment will push structural biology towards new frontiers, deep into the protein universe that is dark and elusive for current structure analysis methods.

Chapter 8

Refining Constraints with Corroborating Evidence

The work presented in this chapter is original to this dissertation.

Own contributions: I conceived and designed the experiments, conceived and designed the algorithm, implemented the algorithm, performed experiments, developed analysis tools, analyzed the data.

Contributions of co-authors: The algorithm presented in chapter 8 was conceived by me (MS), Juri Rappsilber, and Oliver Brock at a research meeting at the University of Edinburgh. OB and JR scientifically advised this work.

8.1 Introduction

In the previous chapters of this thesis, we leveraged two different types of non-local constraints —contact constraints (chapter 5) and CLMS constraints (chapter 7)— to improve the prediction of protein structure. These constraints represent upper Euclidean distance bounds between residue pairs. We demonstrated that these constraints contain valuable information to compute the native structure of a protein.

The origin of these constraints is a measurement or computational prediction process. However, the measurement or prediction processes are not perfectly accurate. Therefore, the constraints contain some degree of noise, i.e. false positives. Since the accuracy of constraint information is critical for structure prediction, improving the accuracy of constraints should also result in improved structure prediction. In this chapter, we devise a novel method to refine constraint information using *corroborating evidence* between constraints.

Most methods derive constraints at a per-constraint basis. For example, cross-link/mass spectrometry based methods measure each cross-link individually. However, if we only consider each constraint individually, we disregard corroborating evidence *between* constraints. Since corroborating evidence is an information source that we did not use to derive the constraint, explicitly leveraging corroborating evidence should increase constraint accuracy.

Figure 8.1 shows an intuitive example of this concept. Consider the task of constructing a city map without having access to any map information (Figure 8.1). Let us also consider that we cannot measure the distance between cities. Instead we only have access to pairwise constraint information between a small number of cities (cities are closer than 2000 km). Assume that the set of available constraints also contains false positives. We now aim to refine this set of constraints by leveraging corroborating evidence. In this example, we know on which continent the cities are located. Leveraging this knowledge makes constraints between cities on the same continent (for example Europe) much more likely than constraints between cities on different continents (for example Europe and South America). Using this information boosts our confidence in constraints that are consistent with the corroborating evidence.

In this chapter, we refine non-local constraint information for protein structure prediction using the concept of corroborating evidence. The key to this approach is the exploitation of corroborating evidence that is specific to the measurement process or prediction method. We use this approach to refine constraint data from CLMS experiments and predicted contacts from EPC-map (introduced in chapter 7 and chapter 5, respectively).

Section 8.2 introduces the data structures and algorithms to encode and leverage corroborating information. This section also introduces the specific types of corroborating evidence that we use to refine CLMS and contact data. Section 8.3.2 details the implementation. In section 8.4, we present the results of the proposed method. We demonstrate that leveraging corroborating information increases the accuracy of CLMS and contact data.

8.2 Overview of the Algorithm

The key idea of our method is that constraint data is typically derived by considering each constraint individually, thereby neglecting any information between constraints. Thus, corroborating evidence between constraints is an additional information source that can be leveraged to refine the data. We now introduce the data structures and algorithms that we use to represent and leverage corroborating evidence. Refer to Figure 8.1 for an graphical overview of the method. For the sake of clarity, Figure 8.1 introduces the concept of the

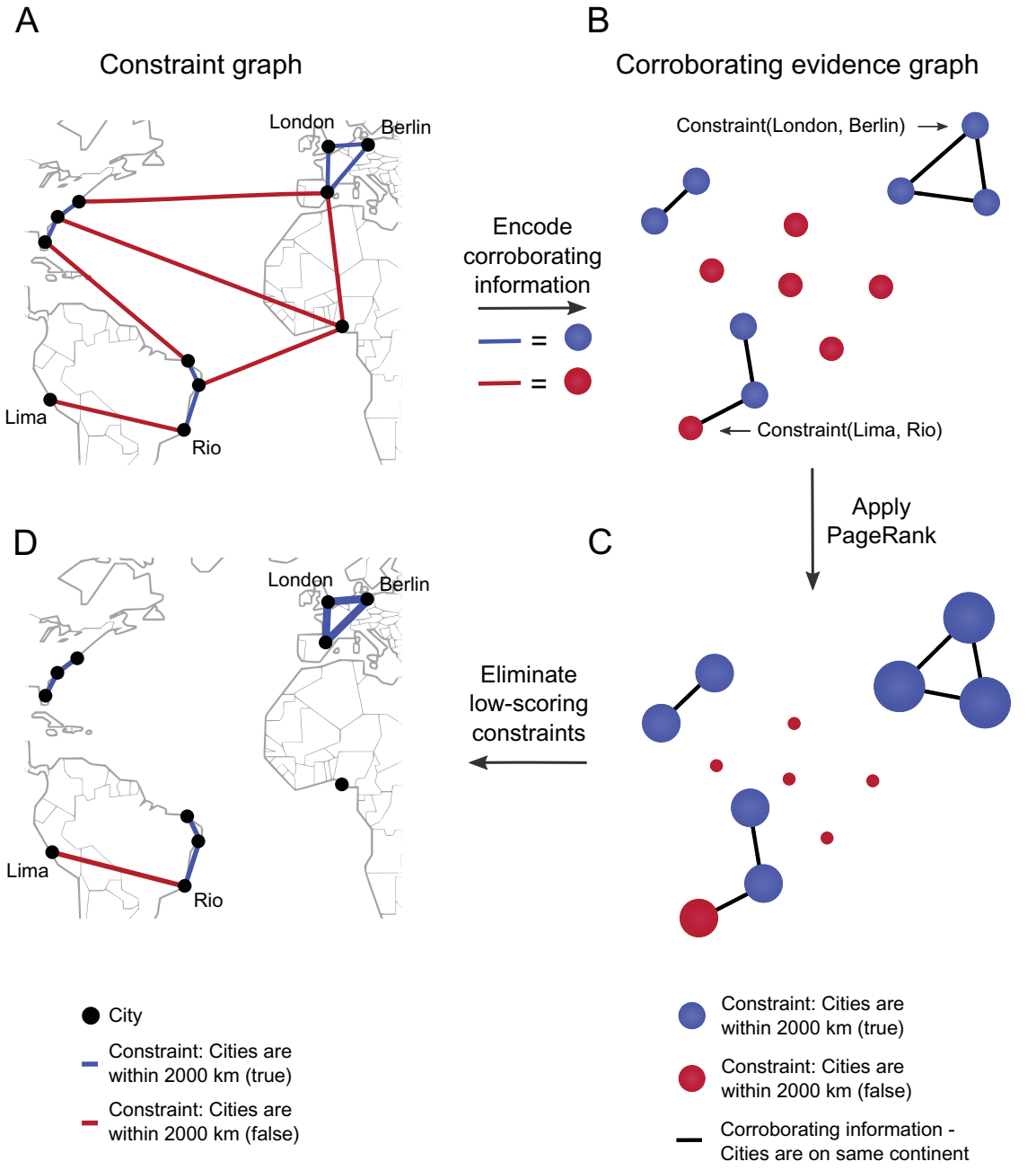

Fig. 8.1 Outline of constraint refinement by leveraging corroborating evidence on a world map toy example. Consider the task of constructing a map by only having access to pairwise constraints between cities. A constraint between a city pair indicates that these cities are within 2000 km distance. We do not have any knowledge of the map beforehand but do know on which continent each city is located. **A**: *Constraint graph* showing the constraints between cities. True positive constraints (the cities are within 2000 km) are shown in blue, false positive constraints are shown in red. **B**: The first step of the algorithm, we construct the *corroborating evidence graph*. In this graph, a node represents a constraint while an edge represents corroborating information between the constraints. In our example, we connect two constraint nodes with an edge if all involved cities are on the same continent. **C**: The graph topology of the *corroborating evidence graph* encodes the corroborating evidence. We now measure the influence of each node in the network by the PageRank algorithm. In this illustration, the size of a node is proportional to its score computed by PageRank. **D**: We now map the corroborating evidence scores back to the original *constraint graph*. The thickness of an edge is proportional to the corroborating evidence score of the constraint. In the last step, we filter the constraint set by eliminating low scoring constraints.

proposed method on a toy problem. In this toy problem, we refine the a set of constraints between cities.

The input to our algorithm is a set of pairwise constraints between objects, the *constraint graph* (Figure 8.1A). In the first step, we encode corroborating information into the *corroborating evidence graph* (Figure 8.1B). In this graph, nodes represent constraints and edges represent the corroborating evidence between these constraints. Note that the corroborating information is problem-specific. We then apply the PageRank algorithm on the *corroborating evidence graph* (Figure 8.1C). The algorithm propagates corroborating evidence through the graph and results in a steady state. In this steady state, the score of a certain constraint node is proportional to the amount of corroborating evidence pointing to that constraint. This effectively rescores the constraints, which we map back to the original *constraint graph* (Figure 8.1D). Note that the algorithm filters the constraints by re-scoring and elimination of low-scoring constraints. Thus, this procedure reduces the number of constraints that we used as input to the algorithm.

Importantly, the proposed algorithm requires corroborating information that is specific to the constraint data that we aim to refine. In the following two sections, we introduce the specific corroborating information that we use to refine CLMS and contact data.

8.2.1 Corroborating Evidence in CLMS Data

A cross-link between two residues indicates that this residue pair in the folded protein is within a certain distance threshold that is proportional to the cross-linker length. However, due to the technical limitations of the mass spectrometric data acquisition, data analysis, and conformational flexibility of the protein in solution, we need to accept some degree of noise in high-density cross-link data (see sections 4.3.3 and 7.2.3 for details).

We now give three examples of factors that result in noisy CLMS data. Note that there might also exist further unknown factors that contribute to the noise in CLMS data.

Noise from false positive cross-link matches: We determine cross-linked peptides with a false discovery rate (FDR) analysis. Since we cannot define an absolute score cutoff of the peptide spectrum matches, we estimate a score cutoff by also scoring known wrong peptide sequences (decoys). Thus, at a certain false discovery rate of $x\%$, we need to assume that our data contains at least $x\%$ false positive cross-link matches. For details on the FDR analysis, please refer to section 4.3.3.

Noise from uncertain site assignment: Recall that we use a sulfo-SDA cross-linker that contains an NHS-ester on one side and a diazirine on the other side. While the NHS-ester

can only react with lysines and to a lesser extent with S/T/Y residues, the UV activation of the diazirine releases a reactive carbene species that reacts with any amino acid. The broad specificity of the cross-linker makes the data analysis difficult. If we identify a peptide pair, cross-linked by sulfo-SDA, we do not know which peptide reacted with the unspecific carbene species. Thus, a cross-link between peptide 1 and peptide 2 could have formed between any pair of amino acids due to the unspecific reaction. In this case, we would need further mass spectrometric evidence, such as fragmentation of both peptides, to pinpoint the exact cross-link site. As a result, the cross-link data has uncertain site assignments. See section 4.3.2 for an explanation of peptide fragmentation to determine cross-link insertion sites and section 7.2.1 for details on the sulfo-SDA cross-linker.

Long-distance cross-links by conformational flexibility: Long-distance cross-links are links that exceed our estimated 20 Å upper distance bound (see section 7.2.3). The conformational flexibility of the protein in solution results in a heterogeneous conformational mixture. Thus, we might cross-link a region of the protein that is close in space in a particular (possibly minor) conformational species but far apart in the crystal structure. Note that this effect might result in long-distance cross-links, even if there were no errors in data acquisition, data analysis, and site assignment.

However, the CLMS data contains also corroborating evidence between cross-links, which we exploit to reduce noise and therefore increase accuracy. We now introduce the type of corroborating information that we use to refine high-density CLMS data.

Given that one side of the sulfo-SDA cross-linker is unspecific, this site of the linker should react with multiple neighboring residues that are spatially close during independent cross-link events. This results in high link density in the cross-linked region. Thus, we assume that true cross-link sites are frequently supported by neighboring cross-link detections. In contrast, cross-link detections from false positive peptide spectrum matches should rather appear in isolation under the assumption that the probability of observing a wrong spectrum match is uniform along the protein chain (Figure 8.2A). We model this effect by connecting cross-link nodes if the two cross-links connect the same sequence region within a specified sequence separation difference Δ (Figure 8.2B). Note that this Δ accounts for uncertain site assignment because we cannot always exactly pinpoint the linked residues, either. We then reinforce this signal by connecting cross-link nodes that share a common neighbor (Figure 8.2C). This results in dense connections in regions with high corroborating information which increases the propagation of information in that region.

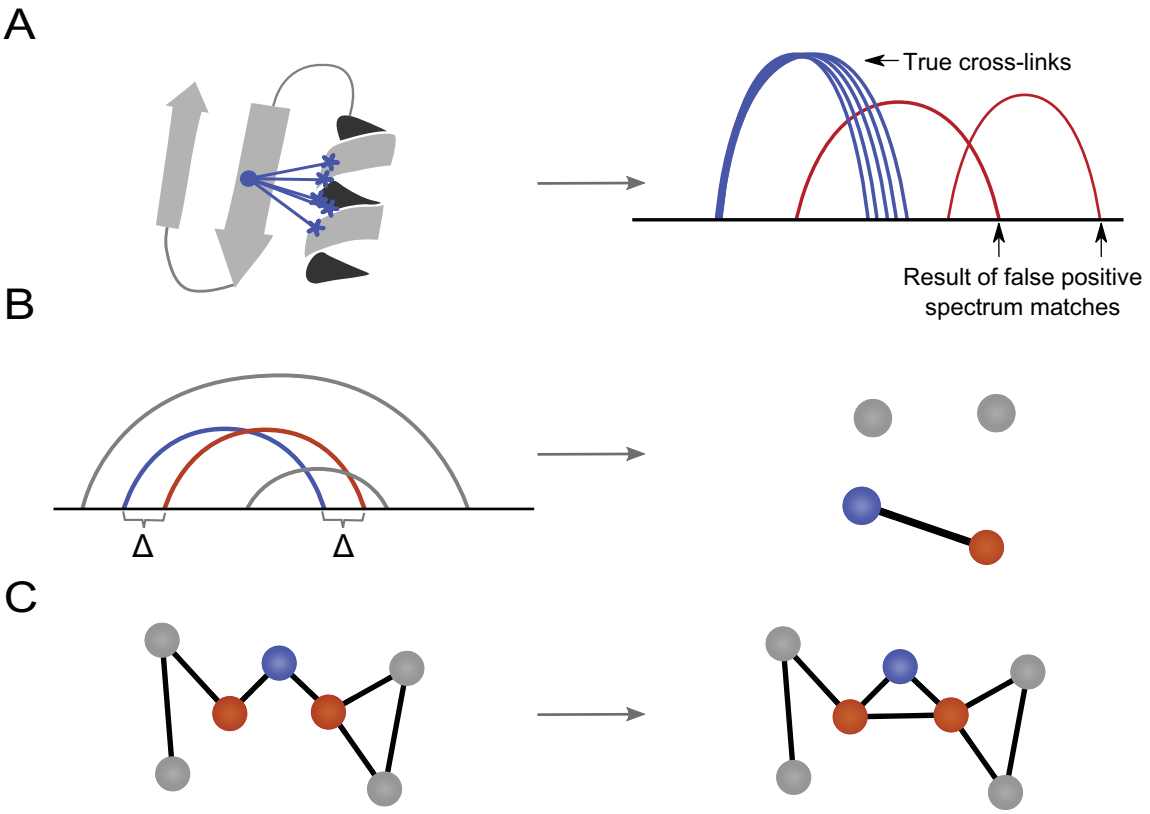

Fig. 8.2 Corroborating evidence in CLMS data. **A**: Because of the unspecific reactivity of sulfo-SDA, we expect to observe multiple neighboring cross-links in a spatially close region of the protein. Thus, a high cross-link density in a region is indicative of true cross-links. In contrast, cross-links that are the result of wrong peptide spectrum matches should be isolated under the assumption that they are uniformly distributed over the protein sequence. **B**: Two cross-links are more likely to be correct if they link to the same residues within sequence separation Δ. In this figure, this condition holds for the blue and orange cross-link. Consequently, these cross-link nodes will be connected by an edge in the corroborating evidence graph. **C**: Two cross-links are more likely to be correct if they share a node in direct neighborhood in the corroborating evidence graph (transitivity). The cross-link nodes under consideration (orange) share one node in their neighborhood (blue). Due to transitivity, the red nodes will also be connected by an edge.

8.2.2 Corroborating Information in Contact Data

Protein contacts are usually defined between residue pairs with a C_β atom distance below 8 Å. Because of the tight upper distance of a contact, the distribution of contacts is the result of the underlying structure. Protein structures are not random but show regular patterns such as α-helices and β-sheets. Thus, if we assume the presence of a contact, it should corroborate other contacts that are in line with our knowledge about the pattern regularity in protein structures. This is illustrated in Figure 8.3 for a contact between α-helices and β-strands. For example, given a helix-helix interface, contact $C(i,j)$ and $C(i+2,j)$ contradict each other because the residue $i+2$ points away from the helix interface (Figure 8.3A). In contrast, the contacts $C(i,j)$ and $C(i+4,j)$ satisfy the physical constraints imposed on protein structure (specifically, helical structure). Thus, the contacts $C(i,j)$ and $C(i+4,j)$ corroborate each other and we connect the nodes of contacts $C(i,j)$ and $C(i+4,j)$ with an edge in the corroborating evidence graph. Given contacts between other secondary structure pairs such as β-strands, certain contacts corroborate each other as the result of the underlying protein structure (see Figure 8.3B).

Given the diverse nature of protein structure, corroborating evidence between contacts should not solely be based on relative sequence positions. Instead, we model the probability for the coexistence of two contacts by their relative sequence shift, conditioned on the secondary structure of the contacting residues (denoted as $P(x,y,|\mathrm{Sec}_1,\mathrm{Sec}_2)$). We obtain the conditional probabilities from observations in high-resolution crystal structures. In the corroborating evidence graph, this probability is reflected by the weight connecting two contact nodes.

8.2.3 Outline of the PageRank Algorithm

The PageRank algorithm is an eigenvector centrality variant. Eigenvector centrality measures the importance of a node by solving the eigenvector equation using the adjacency matrix of the graph (see section 4.1 for details). It was originally designed to capture the relative importance of web pages that are connected by hyperlinks [114, 154]. We now give an introduction to the algorithm that was originally published by Page et al. [154]. The web is modeled by a graph where the nodes represent web pages and the edges represent hyperlinks. Intuitively, the algorithm mimics the behavior of a "random surfer" who navigates through the web by randomly following hyperlinks. PageRank implements the probability of following a hyperlink by the damping parameter α. With probability $(1-\alpha)$, the "random surfer" does not follow a link but instead restarts from a different page. The probability of restarting from a specific page is given by the distribution of the permutation matrix $\mathbf{E}$.

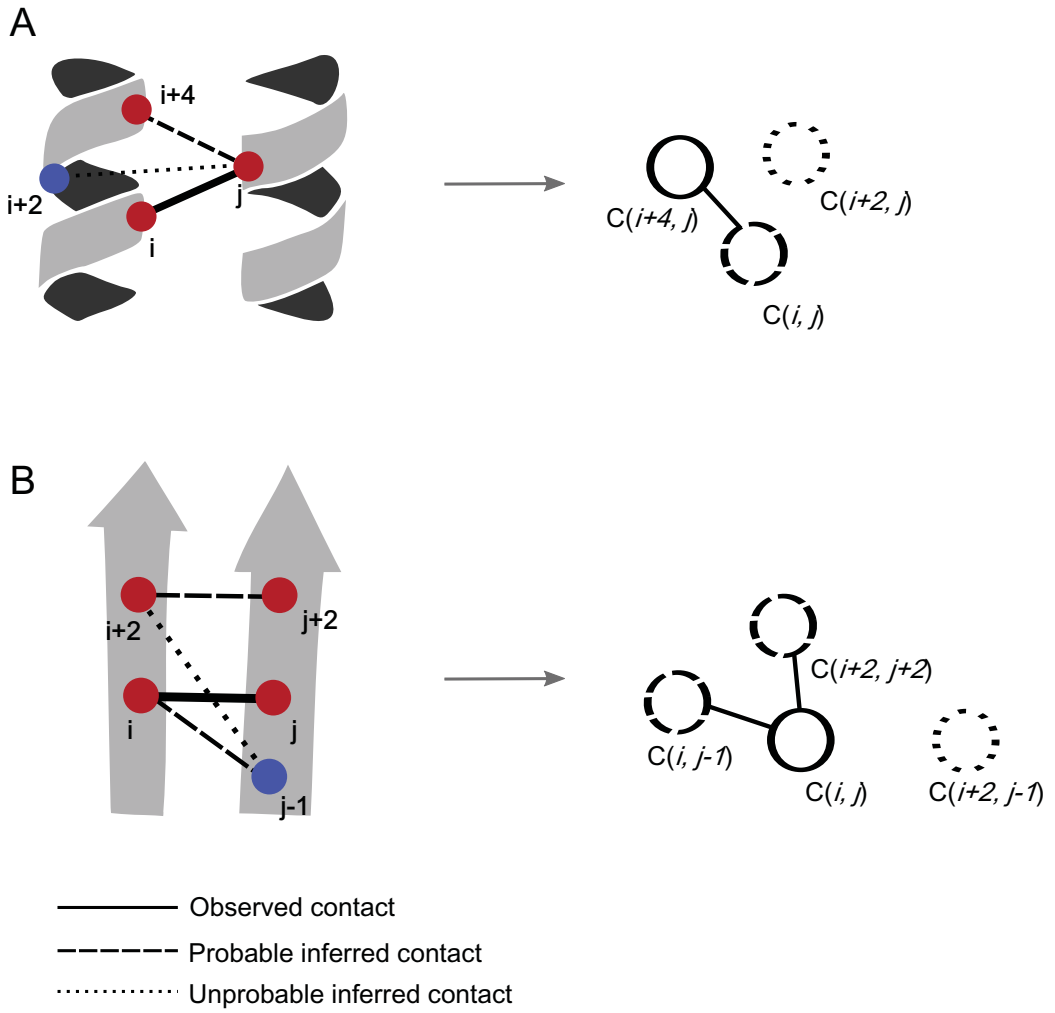

Fig. 8.3 Corroborating evidence in contact data. Contacts are embedded into the 3D protein structure and therefore occur in a pattern that is indicative for this structure. We illustrate this for a helix-helix and a strand-strand contact **A**: Consider a contact between i and j in two adjacent helices. The contact between i+2 and j is unlikely because the i+2 position is on the opposite side of the helix due to the helix geometry. In contrast, the contact between i+4 and j is likely because i+4 is approximately one helix turn above residue i+4. Thus, we connect the contacts $C(i, j)$ and $C(i+4, j)$ with an edge in the corroborating evidence graph. **B**: When observing the sheet-sheet contact $C(i, j)$, there are likely contacts because of the β-sheet geometry. In this example, $C(i+2, j+2)$ and $C(i, j-1)$ would be likely given the β-sheet structure while $C(i+2, j-1)$ would be unlikely. Consequently, $C(i+2, j-1)$ does not corroborate the observed contact $C(i, j)$. Therefore, $C(i+2, j-1)$ is not connected by an edge in the corroborating evidence graph.

This intuition translates into the PageRank equation:

$$\mathbf{r} = \alpha\mathbf{A}\mathbf{r} + (1-\alpha)\mathbf{E}\mathbf{r}.$$

$\mathbf{r} = [r_i]_{Nx1}$ is the column vector with r_i being the PageRank score of node n_i. $\mathbf{A}$ denotes the link adjacency matrix with:

$$a_{ij} = \begin{cases} \frac{1}{N_j} \text{ if } n_i \text{ and } n_j \text{ are connected by an edge; } N_j \text{ is the number of outlinks of } n_j \\ 0 \text{ otherwise} \end{cases}$$

The edges can also be weighted which results in a non-uniform probability distribution in $\mathbf{A}$. $\alpha \in [0,1]$ is the probability of following a link. Finally, $\mathbf{E} = [e_{ij}]_{NxN}$ denotes the perturbation matrix where $e_{ij\in[0,1]}$ is the probability of traversing from node n_i to n_j. The perturbation matrix can be initialized to have a uniform probability distribution.

In this formalization, the PageRank vector $\mathbf{r}$ is understood as the the principal eigenvector of $\alpha\mathbf{A} + (1-\alpha)\mathbf{E}$ [154]. Typical implementations solve this equation by the iterative power method [218].

An important PageRank variant is personalized PageRank [154] where the perturbation matrix $\mathbf{E}$ is replaced by the non-uniform perturbation matrix $\mathbf{F}$ where $\mathbf{F} = \mathbf{f}\mathbf{1}^T$. Here, $\mathbf{f} = [f_i]_{Nx1}$ is the "personalization vector" that represents a non-uniform probability for teleporting to a node and $\mathbf{1}$ a vector with all entries equal to 1. This variant was originally conceived to bias PageRank to a user's personal preference.

In this work, we will encode initial scores of CLMS constraints or contact constraints in the personalization vector. This results in a non-uniform perturbation matrix with a probability distribution that reflects our initial confidence in a constraint/contact. We then construct the corroborating evidence graph which we use as the input to the PageRank algorithm. This rescores the constraints with respect to the initial scores and the corroborating information encoded in the graph.

8.3 Implementation

8.3.1 Implementation of CLMS Data Refinement

The input of the CLMS data refinement is a list of cross-links, sorted by the confidence score of cross-linked peptides that is computed from the peptide spectrum matches (see section 4.3.3 for a detailed explanation of this approach).

We construct the CLMS corroborating evidence graph by representing each link by a node. We connect a node by an edge if the cross-links bind to a residue within a sequence separation $\Delta = 6$ for both ends (see Figure 8.2). In the next step, we connect all nodes that share at least one neighbor in the corroborating evidence graph. We repeat this process two times (two-step transitivity).

The personalization vector **f** contains probabilities equal to the range-normalized cross-link confidence score. We run PageRank on the corroborating evidence graph with a damping parameter of $\alpha = 0.85$ (estimated by cross-validation, see below). We estimate all parameters by leave-one-out cross-validation on the eight proteins for which we have high-density cross-link data available.

8.3.2 Implementation of Contact Data Refinement

The input to contact data refinement are the predicted contacts from EPC-map (see chapter 5), sorted by their prediction score. Predicted contacts contain false positives that we aim to eliminate by leveraging corroborating information between contacts.

We construct the contact corroborating evidence graph by representing each contact by a node. We start with the highest scoring contact node (of contact $C(i,j)$) and connect edges to other contact nodes that are within eight positions around $C(i,j)$. The weight of the edge is given by the probability of $P(x,y,|\mathrm{Sec}_i,\mathrm{Sec}_j)$, where Sec_i and Sec_j are the secondary structure assignments of residue i and j, as predicted by PSIPRED [86]. The arguments x and y denote the shift in sequence positions around $C(i,j)$ of the second contact node.

We obtain the probability tables $P(x,y,|\mathrm{Sec}_i,\mathrm{Sec}_j)$ from the analysis of contacts in the EPC-map training set (see section 5.3.9), by measuring the co-occurrence frequencies of contacts, conditioned on secondary structure and their relative shift in the sequence. For an observed contact $C(i,j)$ we increment the counter by one if the C_β atom distance of residues i and j is below 8 Å. If the distance between $C(i,j)$ is slightly larger than 8 Å, we still increment the counter but only by $e^{-\frac{(d_{ij}-8)^2}{0.05}}$. This still leads to the consideration of residue pairs with a distance d_{ij} slightly larger than 8 Å. This reduces noise when computing probabilities because a residue pair distance that is just outside the allowed contact range would not count otherwise. For example, a residue distance of 8.1 Å still contributes 0.82 "counts" (instead of a full count if it where smaller or equal than 8 Å).

For contact data, the personalization vector **f** contains range-normalized output scores from EPC-map. We use a damping factor of $\alpha = 0.4$ (estimated by cross-validation, see below). Presumably, the damping factor for contact refinement is different from CLMS refinement because of the different network topology in both cases and the relative influence

of the input personalization scores. Contact prediction methods predict a probability for every possible contact, but since the precision of the entire contact map is usually low, only the top scoring contacts are considered. This leads to another parameter β that denotes the top contacts that we consider to construct the corroborating evidence graph, where L is the length of the sequence. In our experiments, we use $\beta = 2$. We estimate all parameters by 10-fold cross-validation on EPC-map_train.

8.4 Results and Discussion

8.4.1 Refinement of CLMS Data

Figure 8.4 shows the results for refining CLMS data with corroborating information. At the time of this dissertation, we had high-density photo cross-linking data sets available for eight proteins: Green fluorescent protein (GFP), human-serum albumin (HSA), glutathione S-transferase (GST), maltodextrin-binding protein (MDP), and four CASP11 targets (Tx767, Tx781, Tx808, and Tx812). The data sets were made available from the Rappsilber laboratory.

Recall that the algorithm re-ranks the input CLMS data and eliminates low-scoring links. For experiments on the CLMS data, we eliminate the lowest scoring 50% cross-links to quantify the effect of this re-ranking.

The CLMS accuracy improves for five out of eight cases (accuracy does not improve for Tx767, GST, and Tx781). Note that the initial CLMS accuracy for MDP is low. Corroborating evidence improves the accuracy for MDP, but it is still below 0.3. CLMS data for HSA improves drastically by corroborating evidence (+11.3%). On average, our approach improves the mean accuracy of the top 50% cross-links by 3% (Figure 8.4A). Another measure to quantify the discriminatory power of a score to rank cross-links is the receiver operator characteristic (ROC). We summarize the ROC curve using the area under ROC curve (AUROC). This area is proportional to the probability that a uniformly chosen positive example ranks higher than a uniformly chosen negative example. Corroborating evidence increases the AUROC by 2% from 0.70 to 0.72 (Figure 8.4B). For five out of eight cases, the ROC curve rises earlier for the CLMS confidence score than for the corroborating evidence score (Figure 8.5). Although the CLMS confidence score ranks the top cross-links better than the corroborating evidence score, the latter improves the ranking overall. This partially explains the relatively small increase in AUROC when using corroborating evidence. However, maximizing the number of cross-links is beneficial for structure prediction. Therefore, the ranking power of the top cross-links is not relevant for structure prediction. For other

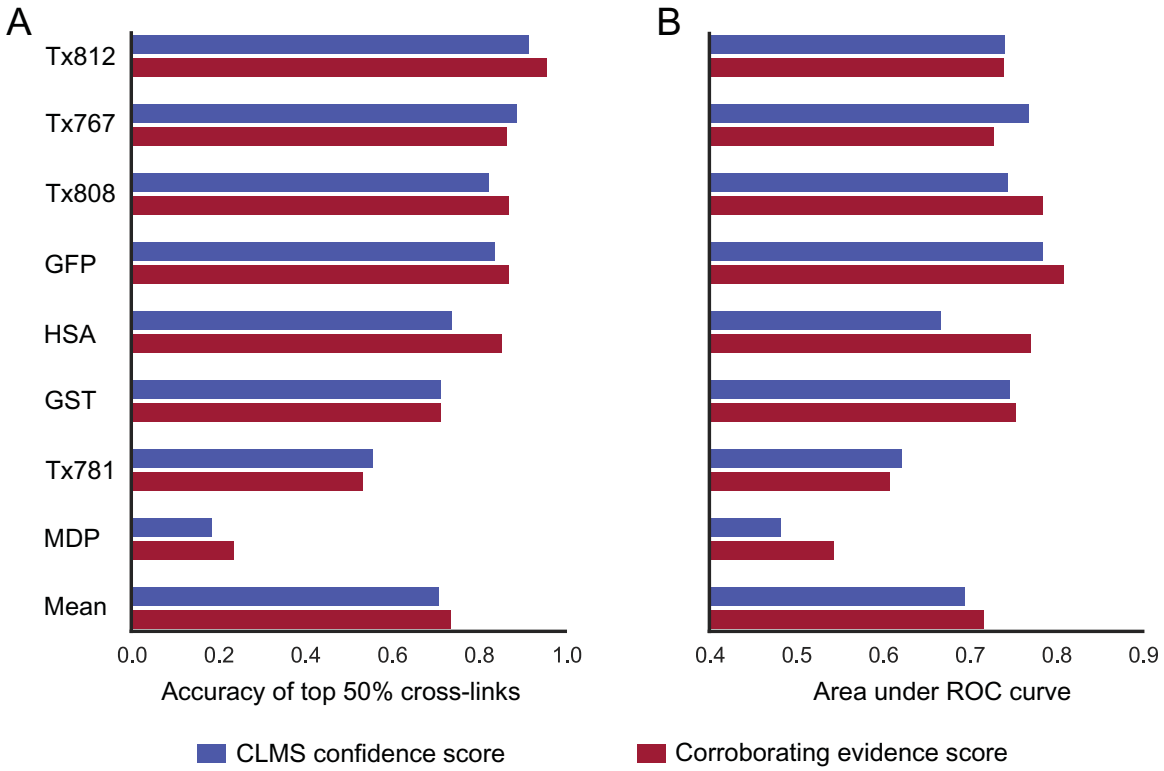

Fig. 8.4 Refinement of CLMS data with corroborating information. Refinement of CLMS data from green fluorescent protein (GFP), human serum albumin (HSA), glutathione S-transferase (GST), maltodextrin-binding protein (MDP), and four CASP11 targets (Tx767, Tx781, Tx808, and Tx812). **A**: Cross-link accuracy (cross-links under 20 Å distance) by CLMS confidence score and corroborating evidence score. We assess the accuracy of the top scoring 50% cross-links because corroborating evidence only re-ranks CLMS data and therefore the accuracy of the entire cross-link set is constant. On average, re-ranking with corroborating evidence improves the CLMS accuracy by 3% (from 0.71 to 0.74). **B**: Area under ROC curve (AUROC) for links ranked by confidence score and corroborating evidence score. On average, corroborating evidence improves the AUROC by 2% (from 0.70 to 0.72).

applications where the accuracy of the top links is important, the CLMS confidence score provides a better ranking.

Corroborating information worsens the ranking for Tx767 and Tx781. Both proteins are CASP11 targets that have been tested in a blind test in which the structure was not known to the Rappsilber laboratory. Tx781 suffered from significant aggregation during shipping from the CASP organizing committee. Thus, the high number of false positives cross-links is probably a result of the aggregated protein sample. We assume that the cross-link pattern for Tx781 deviates from the pattern that would be observed in folded proteins and therefore the corroborating evidence cannot increase the accuracy. For Tx767, we observe a good density (1.46 links per residue) and distribution of cross-links over the structure. However,

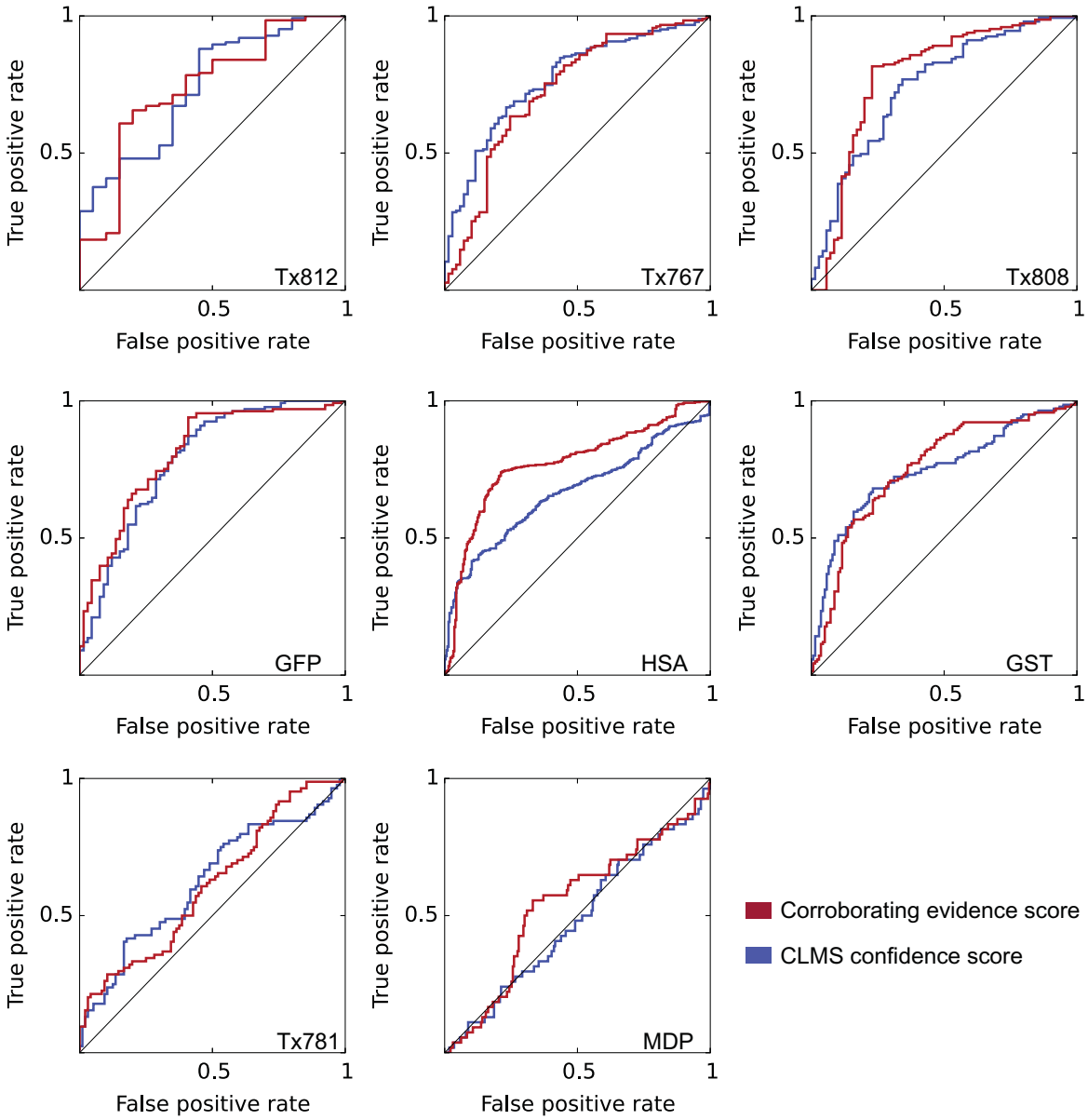

Fig. 8.5 Receiver operator characteristic (ROC) for refined CLMS data. For five proteins, the CLMS confidence score is better in ranking the top scoring links. This results in an earlier rise of the CLMS confidence ROC curve (blue) compared to the corroborating evidence curve (red). However, the AUROC value increases for five out of eight proteins.

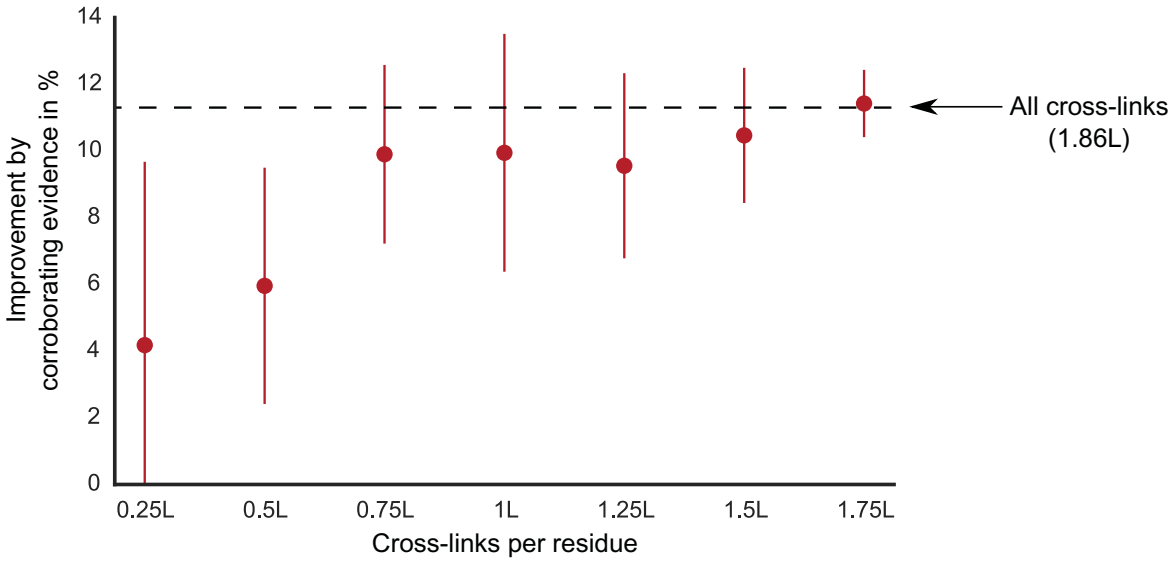

Fig. 8.6 Improvement of link accuracy as a function of CLMS density (results for HSA). In this experiment, we subsample the CLMS data and then assess the improvement in accuracy of the top 50 % links by the corroborating information algorithm over CLMS confidence scoring. The figure depicts the mean accuracy (red points) and standard deviation (red lines) over ten subsamplings at a certain cross-link density. If we sample only few cross-links (low cross-link density) the gain in accuracy is small (see 0.25L and 0.5L). Sampling more cross-links (high density) results in higher accuracy gains. Thus, corroborating information is likely to further boost the quality of high-density CLMS data.

corroborating evidence reduces the accuracy of CLMS data for this protein. The exact reason for this is not clear at this point and we would need to analyze additional examples of failed CLMS refinement.

The accuracy of HSA improves most significantly (+11.3%) while the accuracy of the other proteins (excluding Tx767 and Tx781) only improves up to 5%. HSA has the highest density of CLMS (1.86 links per residue), while the other proteins have a density of around one link per residue. Intuitively, a high CLMS density also leads to more opportunities for cross-links to support each other. Thus, higher cross-link density should result in a larger improvement when using corroborating information.

We tested this hypothesis by subsampling CLMS from HSA and ranking them by corroborating evidence. Indeed, the cross-link density impacts the CLMS accuracy from corroborating information ranking (Figure 8.6). When only 0.25L and 0.5L links are available, the gain in accuracy is much smaller. The improvement is only at the level of the entire cross-link set when we sample at least 1.75L cross-links. However, note that we only performed this experiment on a single protein (HSA) and the impact of cross-link density might vary for proteins with different CLMS distribution or folds. Nevertheless, our results

indicate that corroborating evidence is especially effective for high density CLMS data. Thus, corroborating evidence is an effective and simple strategy to boost high density CLMS data.

8.4.2 Refinement of Contact Data

In this section, we present the results on refining contact data. Specifically, we will refine predicted contacts from EPC-map by corroborating evidence. In the previous section, we leveraged corroborating information from the specific measurement process. In this section, we leverage a different kind of corroborating information that is based on prior knowledge of protein geometry.

Because residue pairs in contact are close in distance (C_β atom distance below 8 Å), a physical contact map should exhibit a pattern that is indicative for the underlying regularities of protein structure. In particular, if we observe a specific contact $C(i, j)$ we would expect a contact pattern that is specific to the secondary structure of residues i and j (see section 8.2.2). We computed the probabilities of contact-contact patterns, conditioned on the secondary structure of i and j, from the EPC-map training set (see section 5.3.9). Figure 8.7 shows the contact co-occurence for helix, strand, and coil secondary structures. These co-occurence matrices form the basis for the corroborating information that we use to refine predicted contact data.

For each protein, we construct the corroborating evidence graph by taking the top 2L contacts. We construct the corroborating evidence graph by iterating over the nodes n_i (contacts) and form an edge to each other node n_j that is within a window of 8 residues around the contact of n_i. The weight of the edge is equal to the probability from the co-occurence matrices (Figure 8.7). We then re-rank the contacts by running PageRank on this graph.

Figure 8.8 shows the result of refining EPC-map contacts on the EPC-map_test data set. Overall, the refined contacts are more accurate than EPC-map contacts for any number of top analyzed contacts. The absolute improvement in accuracy for the top $L/5/L/2/1L/1.5L$ contacts is 1.8/2.3/2.1/1.0% for medium+long-range contacts (sequence separation larger than 11 amino acids) and 2.4/2.3/0.5/% for long-range contacts (sequence separation larger than 23 amino acids).

The refinement does not improve the top $1L$ and $1.5L$ contacts because our algorithm takes the top $2L$ medium+long range contact list as input. This input contact list might contain fewer than $1L$ or $1.5L$ long-range contacts. Since our algorithm only re-ranks the input contacts, we evaluate exactly the same long-range contacts in these cases. However, the changed ranking does not influence the accuracy of a contact set that consists of the same contacts. Nevertheless, our results show that refinement increases the accuracy of the top $L/5$

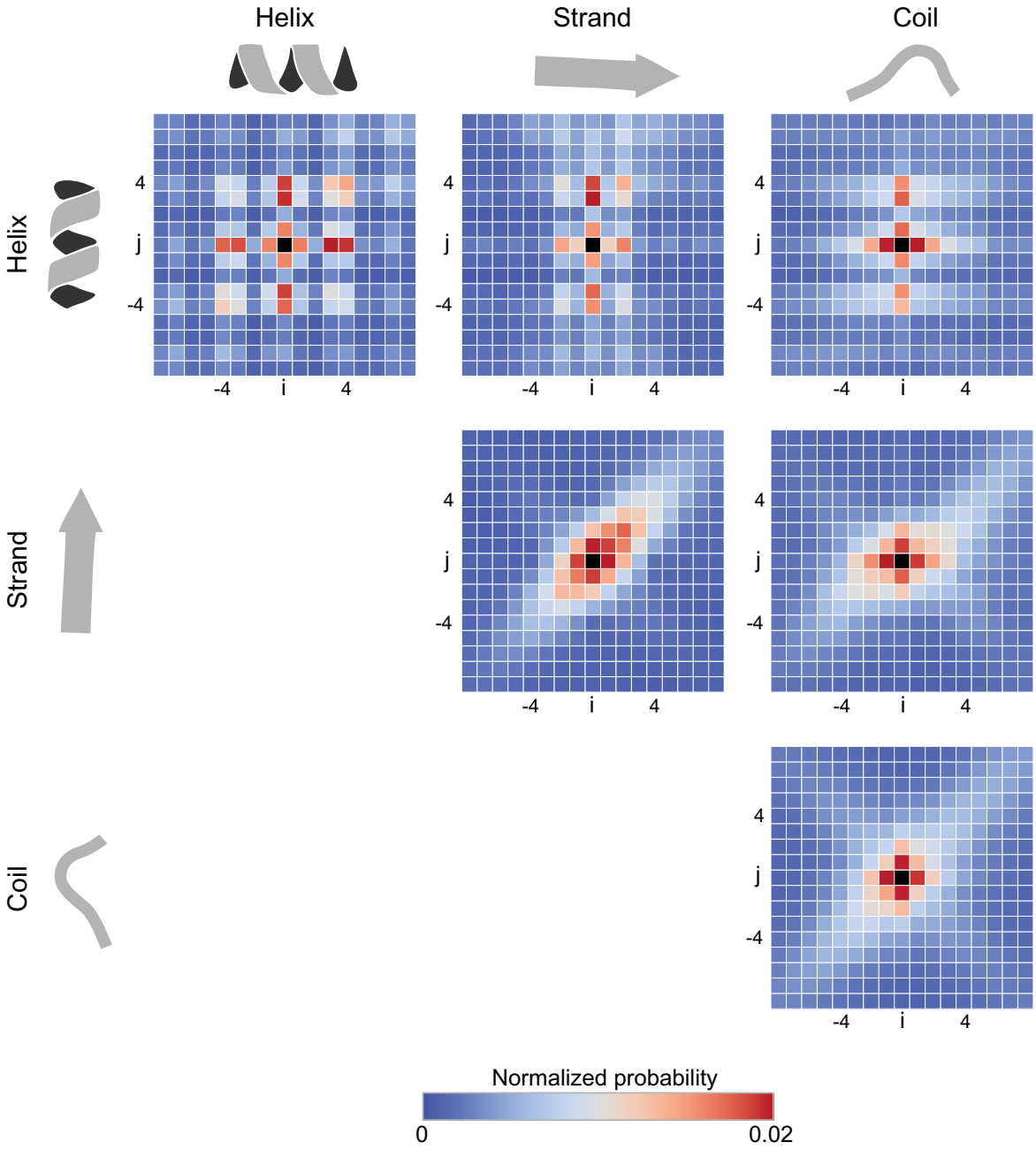

Fig. 8.7 Co-occurence matrices for contact refinement. We compute co-occurence matrices by centering on a contact and computing the co-occurence probability of another contact up to eight residue positions away in i and j direction. We condition the co-occurence matrices on the secondary structure of the residues of the centered contact $(C(i, j))$. This results in nine contact matrices for helix, strand, and coil secondary structure assignments. We omit the lower diagonal because they essentially correspond to the rotated matrices of the upper diagonal (Helix-Sheet is a rotated Sheet-Helix matrix). The co-occurence patterns are therefore characteristic to the underlying secondary structure of the centered contact. For example, if we observe a helix-helix contact $C(i, j)$, the most probable contacts are three or four residues away in either i or j direction, which is consistent with the helix geometry of 3.6 residues per turn. We compute the co-occurence matrices on 742 proteins from the EPC-map_train data set (see section 5.3.9).

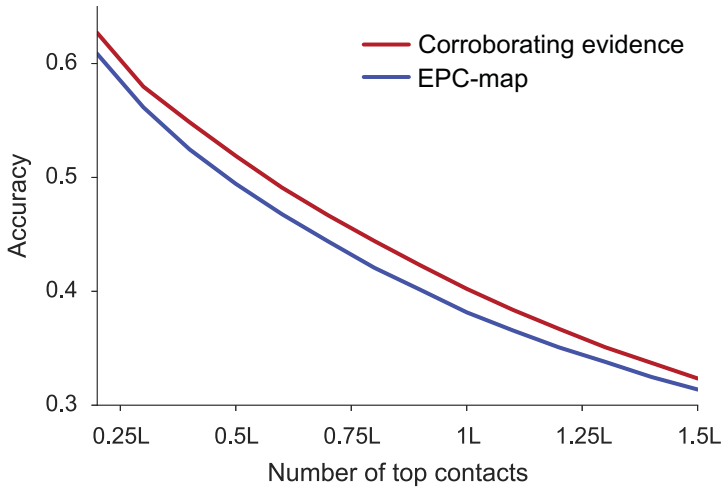

Fig. 8.8 Comparison of EPC-map and refined contact accuracy. This figure shows the contact accuracy as a function of the top xL analyzed contacts, where L is the length of the protein. This plot shows the medium+long-range accuracy (sequence separation larger than 11 residues). Corroborating evidence consistently refines EPC-map contacts over all top contact cutoffs. Results for 132 proteins from the EPC-map_test data set (see section 5.3.9).

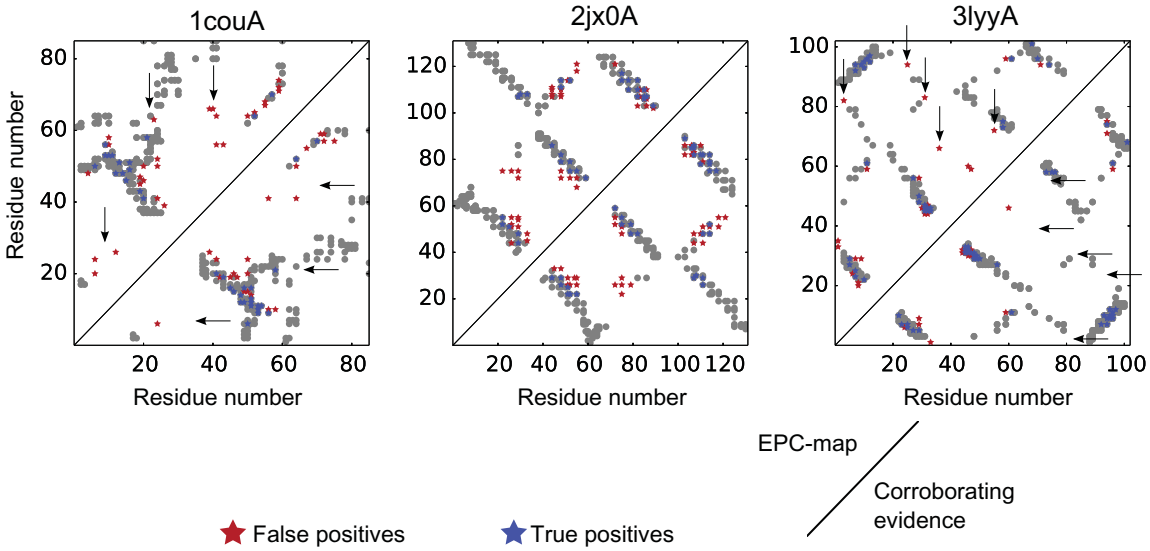

Fig. 8.9 Examples of refined contact maps (top $L/2$ contacts). The upper triangle shows EPC-map contact maps; the lower triangle the contact maps refined by corroborating evidence. Arrows indicate parts of the contact maps that are significantly refined by corroborating evidence.

and $L/2$ long-range contacts. It might be possible to modify the algorithm to refine a larger number of long-range contacts by using more long-range contacts as input and by using co-occurence matrices that are specific to long-range contacts. Another variant could infer probable contacts based on the input contact pattern, which would effectively add different contacts and go beyond mere re-ranking.

Figure 8.9 shows three examples of refined contact maps. The accuracy of the contact maps increases from 0.381/0.446/0.588 (EPC-map) to 0.5/0.508/0.745 for (corroborating evidence) for 1couA/2jx0A/3lyyA.

In summary, our results suggest that a simple contact co-occurence model in combination with network analysis is already able to refine predicted contact maps.

8.5 Conclusion

In this chapter, we presented an approach to leverage corroborating information to refine structural constraints. Key to our method is the encoding of corroborating evidence between structural constraints in a graph structure. We then use the PageRank algorithm to re-score the structural constraints using corroborating information.

We apply this algorithm to constraints from high-density CLMS experiments and to predicted contacts from EPC-map. For CLMS constraints, we find that the effectiveness of this algorithm is dependent on the cross-link density and that the accuracy of high-density cross-links can be significantly improved. For predicted contacts, the gain in accuracy is lower, but nevertheless consistent. Importantly, we increase the accuracy of these constraints by explicitly leveraging the corroborating information between constraints, an information source that is typically neglected.

Theoretically, this algorithm should be generally applicable to refine predictions or measurements in other domains. A requirement for the application of this algorithm is the formulation of corroborating information between data instances.

The striking advantage of our method is that it is fast, simple to implement and generally applicable to any application domain. Thus, we believe that this algorithm has the potential to boost the accuracy of any type of noisy constraint and interaction data.

Chapter 9

Conclusion

Many real-world search problems are high-dimensional and require search in extremely large state spaces. These spaces cannot be searched exhaustively, even with the most advanced computers. Examples from various problem domains suggest that the discovery and exploitation of novel information sources is the decisive factor for effective search and optimization algorithms. Information provides a bias towards the solution. This simplifies the problem tremendously and enables effective search.

In the context of protein structure prediction, we argued that fragments and templates lead to the biggest boost in structure prediction accuracy [146, 147]. We further argued that the field advanced very slowly since the introduction of these methods. We viewed this as evidence that fragments and templates, which ultimately all rely on the statistical analysis of sequence information, might have exhausted their potential. We therefore hypothesized that leveraging novel sources of information is a promising way to advance structure prediction.

In this thesis, we contributed to protein structure prediction by leveraging *novel* sources of information. We devised several algorithms to leverage the following three information sources for protein structure prediction:

1. A graph-based machine learning algorithm to leverage physicochemical information from *ab initio* protein structures and energy functions

2. Information from high-density cross-linking/mass spectrometry, a novel type of experimental data

3. Corroborating information between structural constraints

We demonstrated the value of these information sources for protein structure prediction in extensive experiments.

We now summarize the main findings of this work and their implications (section 9.1). We also discuss the limitations of this study (section 9.2). Afterwards, we recommend future research directions (section 9.3) and comment on the current state and the future in protein structure prediction (section 9.4). Section 9.5 concludes this thesis.

9.1 Summary of Main Findings

Physicochemical Information Improves Contact and *Ab Initio* Structure Prediction

This thesis devised EPC-map, an algorithm that achieves high residue-residue contact prediction accuracy by combining evolutionary information from multiple sequence alignments and physicochemical information from *ab initio* structure prediction decoys. EPC-map leverages physicochemical information by a graph-based machine learning algorithm.

We demonstrated the effectiveness of EPC-map in controlled and in the *blind* CASP11 experiment. In controlled experiments, EPC-map outperformed state-of-the-art residue-residue contact prediction methods and reached 53.2% accuracy on 528 proteins for the top $L/5$ long-range contacts (chapter 5).

In CASP11, EPC-map ranked second for medium+long-range contacts (sequence separation > 11) and fifth for long-range contacts (sequence separation > 23). Note that none of the top five CASP11 contact prediction groups used pure evolutionary methods [33]. We view this as further evidence that physicochemical information significantly contributed to the CASP11 performance of EPC-map (chapter 6).

In addition, EPC-map increased the *ab initio* structure prediction performance of our tertiary structure prediction server RBO Aleph. In the *ab initio* structure prediction category, RBO Aleph ranked first by average Z-score > 0 and third by sum Z-score > -2 (ranking based on first submitted models and assessors' formula, see chapter 6).

Our approach has further implications: An important feature of EPC-map is that we extract physicochemical information directly from *ab initio* decoys. Thus, we extract information about the native structure (in the form of contacts) directly from search space samples. Therefore, EPC-map accumulates information *during* search instead of exploiting information that is available *before* search. We discuss the potential of this finding in section 9.3

Physicochemical Information Is Effective Even When Only Few Sequences Are Available

EPC-map extracts physicochemical information from *ab initio* structure prediction decoys. Since *ab initio* structure prediction is universally applicable to any protein, this source of information is always available. We demonstrated that EPC-map is particularly effective when only few sequence homologs are available, a critical information source for many contact prediction methods. EPC-map is much less dependent on this information source, and therefore more robust in cases in which this information is not available (chapter 5).

The good performance of EPC-map in CASP11 further supports this finding. The protein targets in CASP11 usually resemble especially hard modeling cases with little available sequence information [145] (chapter 6).

The Exploration-Exploitation Problem Is Not Solved in Contact-Guided Structure Prediction

We find that EPC-map contacts improve the GDT_TS of the top scoring model of the RBO Aleph pipeline in combination with model-based search (MBS), but not in combination with Monte Carlo sampling as implemented in the Rosetta protocol. Nevertheless, Monte Carlo sampling leads to higher decoy diversity and infrequently generates decoys closer to native. This finding indicates that the exploration-exploitation problem is not solved in the context of contact-guided structure prediction. Thus, novel algorithms need to find a better balance of the exploitation of constraints and the exploration of the conformational space (chapter 6).

Incompatible Components Are an Significant Impediment to Structure Prediction

We find that the components in a structure prediction pipeline interact in unexpected ways and that an improved component does not necessarily translate into an improvement of the entire pipeline. If this is also the case for other pipelines, this would point to a significant obstacle to the community effort to advance structure prediction. Understanding these interactions of components, or at least ensuring compatibility between them, might be vital to advance protein structure prediction (chapter 6).

We believe that this finding has important implications. The primary question behind assessing the state of the art of structure prediction methods is to reveal which methods work and *why* they work. Our findings about component interactions in RBO Aleph, if they generalize to other pipelines, suggest that only assessing entire pipelines is not sufficient to understand the impact of all components. Thus, we think that novel structure prediction com-

ponents need to be analyzed in isolation *and* in the context of complex structure prediction pipelines. This is critical to deepen our understanding and ultimately advance protein structure prediction. A short term solution could be to disseminate re-tuneable and re-trainable algorithms that make the integration of component methods into different pipelines easier.

High-Density CLMS Data Enables Determination of Protein Structure

We combine high-density cross-linking/mass spectrometry data (CLMS) with conformational space search to reconstruct the structure of the domains of human serum albumin (HSA) with good resolution (2.5/4.9/2.9 Å RMSD for the best-of-five structures of domains A/B/C). This is the first instance of a high-density CLMS/*ab initio* structure prediction hybrid method. State-of-the-art *ab initio* structure prediction guided by standard BS3 cross-links is not able to reliably predict the structure of this protein. This demonstrates that combining high-density CLMS data with conformational space search is a promising avenue towards structure determination (chapter 7).

High-Density/Noisy CLMS Data Is Superior to Low-Density/Accurate CLMS Data in Structure Determination

We found that high-density CLMS data is more effective than low-density CLMS data for protein structure determination, even if the high-density data set is much noisier. Low-energy structure ensembles (1% energy percentile) generated using CLMS data at 10/20% false discovery rate (FDR) have a lower RMSD than low-energy ensembles ensembles generated at 1/5% FDR (chapter 7).

This implies that future research efforts should: 1) Aim to further increase CLMS density, even at the cost of noise. 2) Explicitly address noise to optimally leverage CLMS data.

CLMS Enables Structure Determination in Complex Biological Environments

We demonstrated the reconstruction of HSA domain structures in their native environment, human blood serum (RMSD 3.5/5.2/3.8 Å for the best-of-five structures of domains A/B/C, chapter 7). X-ray crystallography or NMR are unable to probe proteins that cannot be extracted from their native environment, many of which are critical for our understanding of biology or associated with disease. Since our CLMS-based hybrid method is able to analyze the structure of proteins in their native environment, this method is poised to investigate the "dark matter" of the protein universe, protein systems that are out of reach of current structure determination methods.

Additionally, this is an important stepping stone towards observing conformational changes in response to molecular processes in the cell. We discuss this potential future direction in section 9.3.

Leveraging Corroborating Information Is a Simple Way to Refine Structural Constraints

Structural constraints are typically derived on a per-constraint basis, which neglects any corroborating information between them. We showed that explicitly leveraging corroborating information refines high-density CLMS constraints and predicted contacts. This algorithm is fast, simple to implement, and can be applied to any problem domain. Thus, we believe that this algorithm has the potential to boost the accuracy of any type of noisy constraint and interaction data, eventually becoming a standard post-processing step in many application domains.

9.2 Limitations

Limitations of EPC-map

EPC-map generates *ab initio* structures to extract physicochemical information. However, this step is computationally expensive. Rosetta needs approximately 10 minutes to generate a decoy of a protein with 250 amino acids, which results in seven CPU days for 1000 decoys. We parallelize Rosetta on 100 nodes which results in 100 minutes for decoy generation for a 250 residue protein [1]. The limitation of high computational cost might render EPC-map unsuitable for proteom-wide analysis of protein contacts. However, even low-cost commodity clusters have sufficient CPU power to run EPC-map for many practical applications. These computational resources are probably already available for research groups that work on *ab initio* structure prediction, which is the target audience of EPC-map. In fact, using predicted contacts might even speed up *ab initio* structure prediction because fewer decoys need to be generated to obtain native-like structures. In any case, the interested user can obtain EPC-map contacts from our web service: http://compbio.robotics.tu-berlin.de/epc-map/.

Another limitation is that contact prediction with EPC-map is less effective if all generated decoys are of low quality (in terms of RMSD to native). If decoys do not contain (at least some) native contacts, no native contacts can be predicted by EPC-map. Since free modeling in CASP focuses on very difficult protein targets, this is probably the main reason for the

[1] The other steps in EPC-map, feature generation and contact prediction with the SVM ensemble, are computationally cheap (less than five minutes on a single CPU).

performance gap of EPC-map in our controlled experiments and in CASP11. However, we can address this problem by using alternative methods for decoy generation that are able to sample near-native structures of large proteins with complex topology [23, 99, 172]. Thus, this does not represent an inherent limitation of EPC-map.

Limitations of High-Density CLMS-Based Hybrid Methods

The main limitation of our high-density CLMS study is that we only demonstrated it on one protein, human serum albumin. Thus, some of the findings in this study, such as the effectiveness of various model selection methods, might not generalize to other proteins. However, the Rappsilber group successfully demonstrated that high-density CLMS can be applied to other protein by blind-testing their method in CASP11 (Schneider et al., submitted to *Proteins*). This testing revealed further technical limitations of the CLMS approach. Long stretches in the sequence without trypsin digestion sites result in large peptides that cannot be detected by mass spectrometry. Thus, these stretches will not contain cross-links. Furthermore, the cross-link upper distance bound of 20 Å is too high to pinpoint secondary structure arrangements such as β-sheets (approximately 5 Å distance between individual strands in a sheet).

In addition, one could argue that HSA is not a representative example for structure determination in natural environments because human blood serum is mostly composed by HSA anyways. However, performing this analysis on less abundant proteins is merely a technical issue. Less abundant proteins would simply require more sophisticated enrichment techniques of cross-linked peptides.

We would like to stress that all the discussed limitations are of merely technical nature and can, in principle, be overcome (see section 9.3).

Limitations of Corroborating Information to Constraint Refinement

Since we only use corroborating information to *refine* constraint data, this approach is only effective if the data is already sufficiently accurate. This implies that this algorithm is not suited to fix poor data sets. In addition, our experiments on refining CLMS data suggest that high constraint density is necessary for this algorithm. Presumably, sparse data sets do not contain enough corroborating evidence for refinement. Therefore, corroborating information might be ineffective to refine sparse experimental data, such as EPR [79], sparse NMR data [166], or CLMS data from specific cross-liker reagents [231].

Corroborating information also needs to be specifically devised for each application domain. This represents another potential limitation, because the knowledge to devise

corroborating information might not be available in each domain. In such cases, the algorithm will not be applicable. On the other hand, our method might serve as a tool to test the scientific understanding in a specific domain, because corroborating information based on valid assumptions should result in accurate constraints.

9.3 Future Work

We now discuss future research directions to advance the projects presented in this dissertation. Section 9.3.1 discusses immediate actions to improve the performance of the suggested methods. Section 9.3.2 discusses future research opportunities that arise from the results of this thesis.

9.3.1 Improvements of the Proposed Methods

In this section, we discuss actions to improve the performance of the proposed methods.

Improvements of EPC-map

Graph features: The current implementation of EPC-map uses graph-based features to capture physicochemical information. However, the node and edge label statistic features are quite simple. The expressive power of these features could be improved by considering the underlying graph topology. Neumann et al. [149] addressed this issue with propagation kernels and Gibert et al. [65] derived node and edge label statistics relative to representative instances. These or similar approaches could be tested to increase the expressive power of the graph-based representation in EPC-map.

Combination with other information sources: Since the combination of physicochemical information and evolutionary information proved to be effective for EPC-map, the algorithm would benefit from incorporation of further information sources. Sequence-based machine learning would target proteins that are currently problematic for EPC-map (long, complex topology). Current experiments are underway to investigate the combination of sequence-based machine learning with EPC-map, carried out by Kolja Stahl, a master student at the Robotics and Biology Laboratory. Our preliminary experiments on the EPC-map_test data set suggest that this approach is on par or even better than the current leading method in CASP [89].

Improvements of High-Density CLMS/Conformational Space Search Hybrid Methods

Uncertain site assignments: CLMS data analysis cannot always pinpoint the exact cross-linking site. This problem could be addressed by alternating conformational space search with updates of the cross-link site assignments. This could be realized by implementing a procedure that estimates the most likely site assignment given a protein structure ensemble akin to previous work on NMR-based hybrid methods [136].

CLMS distance bounds: Because we cannot exactly pinpoint the cross-linked atoms and need to account for conformational flexibility, we model sulfo-SDA CLMS constraints with a rather low-resolution distance bound of 1.5-20 Å. Tighter CLMS distance bounds would further restrict the conformational space, eventually leading to lower RMSD models. Meier and Söding [134] recently proposed a two-component Gaussian mixture distribution model to capture alignment errors in low-confidence template hits and proposed a probabilistic approach to combine restraints from multiple templates. We propose a similar approach to estimate tighter distance bounds for CLMS constraint from the observed distances of cross-linked residues in the predicted structure ensemble. We think that alternating between CLMS distance bound updates and structure prediction could improve the RMSD of the resulting models. This approach could also estimate whether a CLMS constraint is likely at all or should be rejected as noise, eventually enabling the use of CLMS data at higher false discovery rates.

This approach could be combined with updating likely site assignments during search (see last point). Alternatively, a unified probabilistic mixture framework could be derived to incorporate uncertain site assignments into CLMS distance bound updates.

Improvements for Leveraging Corroborating Information in Constraint Refinement

Corroborating evidence between peptide spectrum matches: Currently, we leverage corroborating information between cross-links to increase the accuracy of CLMS constraints. However, cross-link insertion sites are estimated from peptide spectrum matches that might contain further corroborating evidence. For example, multiple peptide spectrum matches could support the same or different site assignments. Thus, embedding peptide spectrum matches into a corroborating evidence graph for network analysis could further reduce noise in high-density CLMS data.

Estimating topology-specific contact patterns: Our current model of protein contact co-occurence is averaged over proteins with different folds and topology. We propose that fold

and topology-specific co-occurence patterns would be more informative and lead to higher accuracy gains. However, this would add an extra step of predicting the most likely fold of a given protein sequence [110].

9.3.2 New Research Opportunities

In this section, we discuss new research opportunities that arise from this dissertation. In contrast to the previous section 9.3.1, we here propose novel concepts rather than technical advances. Thus, the technical requirements for these concepts might not be foreseen at the moment. The proposed concepts require major shifts of current computational and experimental paradigms and are therefore long term, high risk, but potentially high impact research endeavors.

A New Paradigm for Information-Guided Search

This thesis suggests that using information is critical for effective protein structure prediction. Current algorithms only use these information sources as constraints to find the most likely structure that fits this information. Thus, the quality of the resulting structures is bound to the quality of the input information.

We propose a conceptional algorithmic framework that actively uses search space samples to *refine and complete* input constraints. Such an algorithm would attempt to actively reason about the quality of the input constraints, to filter false positives, or add additional information that is encountered during search. EPC-map might be the first instance of an algorithm that actively extracts spatial constraint information from decoys *during* search. Usually, conformational space search algorithms only exploit spatial constraints that are available *before* search.

This algorithmic principle is analogous to expectation-maximization algorithms [47]. We aim to modify our belief about the constraint set (expectation step) and then predict structures using these constraints (maximization step). At each iteration, we analyze the decoy ensemble to improve the constraint set. This bootstrap approach should improve constraints and add information after each iteration. Over time, the accumulated information should result in improved structure prediction.

This strategy, however, would require a delicate balance of exploration and exploitation. If we exploit information too strongly, we end up in the same search space region in each iteration and no new information can be discovered. If we explore too much, we only skim information from easily accessible search space regions that uninformed search would maybe also discover. Thus, this algorithm would need the capability to steer search in order to

maximize the accumulation of *valuable* information i.e. information that results in improved near-optimal solution states.

We performed first preliminary, but promising experiments with an instance of such an algorithm. In our experiments, we are able to consistently improve the accuracy of contact constraints over multiple iterations. However, the improved contact accuracy did not translate into improved structure prediction[2]. This evidence suggests that accuracy alone is an insufficient measure of the value of constraints, at least in the domain of protein structure prediction. Thus, finding a criterion that measures the value of a constraint set would be the greatest challenge to devise such an algorithm.

However, if this obstacle is overcome, this class of algorithms could be an effective tool to information-guided search in high-dimensional spaces for various application domains.

CellScope: Probing Protein Structure and Dynamics in the Cell

Probably the highest potential from high-density CLMS comes from its applicability in complex, biological matrices. Theoretically, the approach is even applicable in cellular environments. However, soluble cross-linking reagents do not pass the cell membrane or might be too toxic to allow *in vivo* probing of protein structure.

A solution to this issue is the genetic encoding of photo-amino acids (photo AAs) [40]. This approach replaces certain amino acids with photoactivatable variants. This approach has multiple advantages over soluble cross-linking reagents: Because photo AAs are genetically encoded, the proteins can be probed in their mode of action in the cell. Proteins with genetically encoded photo AAs also pass in-cell quality control and therefore adopt the native structure. Since the amino acid itself forms the cross-linker, we expect much higher spatial resolution using this approach. Eventually, the resolution will be sufficient to pinpoint β-sheet arrangements. Also, this approach will be able to deliver cross-links in the core of the protein and not only along the surface.

Perhaps even more importantly, this approach also enables time-resolved structural measurements in the cell. A high power UV laser can be used for rapid activation of cross-links in a flow cell measurement setup. This would deliver high-density and time-resolved data of track conformational changes inside the cell.

However, this approach also faces several challenges. Data analysis will need to deal with very large amounts of low-intensity, high-noise data obtained by high-density cross-linking/mass spectrometry and with the large space of theoretically possible cross-links of non-selective photo-cross-linkers. Detailed understanding of the data needs to guide database

[2]Interestingly, other studies also suggest that improved contact accuracy does not necessarily translate into improved structure prediction accuracy [89].

search [66]. In addition, uncertainty in the data needs to be addressed with false discovery rate estimation, which might be an extension of the method that was already applied for FDR estimation in this thesis [60].

Since high-density CLMS data alone is insufficient for structure determination, we must complement the data with information from structure and sequence databases (fragments, templates, residue-residue contacts). We expect that high-density CLMS data with high spatial resolution would enable novel methods to retrieve such information using the experimental data, akin to using chemical shifts for fragment selection [166].

The biggest challenge probably comes from the heterogeneous mixture of high-density CLMS data in cells. Especially time-resolved measurements will result in data that is averaged over time and different conformations. We propose an algorithm, quite similar to the expectation-maximization algorithm above, that assigns probability densities to cross-links. The probabilities indicate the likelihood of the link to correspond to a particular conformation or time point. For our current belief of the cross-link distribution, we compute a conformational ensemble that represents the most likely distribution of conformations in space and time. The algorithm could alternate between probability density updates and structure ensemble optimization to compute the most likely ensemble model along space and time dimensions that is consistent with the CLMS data.

We proposed this project, CellScope, to the European Union. Although this is clearly a high-risk endeavor we are convinced that the possibility of a technology that is able to observe proteins at their very place of action, inside the cell, might be of unprecedented value for life-science research and very well worth the risk.

9.4 The Current State and the Future of Protein Structure Prediction

In this section, we comment on the current state of protein structure prediction and lay out our opinion for its development in the future. This section will have the form of a commentary and personal view, rather than a review of the current state of the art.

We first comment on the current state of protein structure prediction, with a focus on contact prediction and *ab initio* structure prediction. We lay out multiple hypothesis that could lead to further impact in the field and discuss how this interplays with other advances in structure prediction. We then discuss future development paths of CLMS-based hybrid methods and the possibility of CLMS-based high-throughput structure determination methods.

Protein Contact Prediction

Evolutionary contacts marked a clear milestone towards routine prediction of contacts that have utility in *ab initio* structure prediction. Evolutionary contacts led to the successful prediction of T0806-D1 in CASP11, the largest protein so far (256 residues) that has been predicted by *ab initio* methods in the CASP context [103, 152]. Here, we lay out two research agendas that might advance protein contact prediction.

Contact prediction in the absence of (many) sequence homologs: Another accomplishment is that novel contact prediction methods use evolutionary and other information sources to maintain high prediction accuracy, even in the absence of many sequence homologs [71, 89, 98, 194, 215]. This thesis also develops such an algorithm that is still accurate if many sequence homologs are absent. Certainly, there will be protein families with rich sequence information that are amendable to evolutionary contact prediction. However, this information might never be available for other protein families due to experimental bias, low evolutionary reuse, and other unknown factors. Many of these proteins form the "dark proteome", the set of proteins for which no structures exist and no templates can be identified [159]. We expect that combination approaches that are effective for protein families with little sequence information will be important to probe the structures of the dark proteome. Contact prediction methods that combine many weak priors from multiple information sources (sequence, evolutionary, physiochemical) are more likely to reach a performance level that will allow routine application in protein structure prediction.

Structure-defining contacts: An interesting study suggest that some structure-defining contacts distill most of the information required for successful protein modeling [177]. This suggests that not all contacts are equally important and predicting redundant or "unimportant" contacts will have little value for protein structure prediction. Current studies also suggest that tuning contact prediction algorithms towards high accuracy at the expense of introducing redundancy might actually decrease the value of predicted contacts for structure prediction [89]. It is clear that we need a deeper understanding of which contacts have utility in structure prediction. Using this understanding, we could then derive contact prediction algorithms that predict contacts that are critical for structure prediction.

Ab Initio Structure Prediction

Predicted protein contact information alone is not sufficient to predict protein structure. Thus, contact information needs to be combined with structure prediction algorithms to utilize

this information for structure modeling. Although contact information has some utility in template-based modeling [96], *ab initio* protein structure prediction will likely benefit from predicted contacts to a larger degree. Some of the top prediction groups in CASP11 utilized contact information and reported the benefit of this information [152].

Representation: *Ab initio* structure prediction will be further improved with the ability to utilize contact information effectively. Currently, the focus of most groups is the development of error-tolerant representations of contacts in the energy function that tolerate false positives. Future research might find even more effective contact representations that further increase contact utility in structure prediction.

Sampling: In addition, there is the question of how to effectively sample protein structures that are consistent with contact information. This is especially important for protein targets with little sequence information: Fragment picking algorithms use sequence profiles, thus a protein target with little sequence information will likely also have poor fragments due to poor profile quality. Results from CASP11 suggests that some of the top prediction groups have an advantage by using a sampling method that is orthogonal to fragments [152, 235]. These methods combine models from multiple *ab initio* models and templates and might compensate for poor fragment quality. We expect that these sampling methods and other sampling approaches that directly sample the topology [99] will be critical for modeling larger proteins with contact information. However, there is another open question: Can we directly sample protein topologies that lie on the manifold that is defined by the contact constraints? Such a sampling algorithm would be much more efficient in search since it would generate a higher fraction of structure models that satisfy the contact information.

Using predicted models as information sources: Another route for using contact information is to filter input contact maps with predicted structure models [2]. This view of using structure models as opportunities to refine contact information, and possibly extracting new information, is very much in line with the research presented in this thesis. However, the current methods seem to be limited to a single step of contact refinement/extraction from predicted structural models. Our own preliminary experiments towards alternating between contact and structure prediction also suggest that going beyond one iteration has limited benefit because no new information is discovered. We suspect that the exploration-exploitation problem in contact-guided prediction is the primary obstacle towards an iterative contact/structure prediction algorithm. Some ways to solve this issue have been proposed in this thesis.

Refinement

In addition to improved contact prediction, the refinement of template-based models advanced during the CASP10 and CASP11 rounds [45, 55, 150, 155]. This increased accuracy of template-based models will increase their value for applied modeling, for example to identify lead compounds and binding modes.

Getting structures into the refinement funnel: Even more importantly, these refinement methods now open up a new "funnel" in the protein structure prediction landscape, in which structure models with reasonable accuracy can be further improved towards high accuracy models. If advances in *ab initio* structure prediction will be able to predict structures with sufficient accuracy to enter the "refinement funnel", this could unlock a strong synergy that leads to high resolution, *ab initio* prediction of protein structures.

Hybrid Methods

By their very nature, hybrid methods benefit from advances in protein structure prediction. In particular, many experimental methods also generate spatial constraint data, similar to contacts, and therefore research in contact-guided structure prediction and hybrid methods is very much aligned. We now discuss the opportunities in CLMS-based hybrid method development and discuss their future impact possibilities.

CLMS-based Hybrid Method Development

This thesis demonstrated the use of high-density CLMS data in *ab initio* structure prediction. Further advancing this research requires developments on the experimental and the computational methods.

Experimental advances: On the experimental side, research efforts should focus on further increasing the density of the CLMS data. Furthermore, it will be increasingly important to ensure an even distribution of cross-links over the protein. These problems can be tackled by exploring digestion strategies with multiple proteases and employing other cross-linking reagents with different chemical specificities.

Computational advances: In protein structure prediction, CLMS data is used as spatial constraints, akin to residue-residue contacts. Thus, many of the computational advances that we proposed for contact-guided predictions might also hold for CLMS-guided structure

prediction. Of course, there are CLMS-specific modifications that might be relevant, as discussed in section 9.3.

Concerted development of CLMS-based hybrid methods: We think that the future development of CLMS-based hybrid methods would tremendously benefit from a concerted development. The principle behind this concerted approach is that certain experimental shortcomings might be compensated algorithmically. For example, detailed understanding of the uncertainty in CLMS data could lead to search algorithms that tolerate higher degrees of noise, which in turn allow the experimental method to generate more cross-links at the expense of higher noise. Furthermore, an increased understanding which regions of a protein need cross-link information to improve structure modeling can guide the development of experimental protocols that are adapted to provide data in these critical regions.

Future Possibilities for CLMS-based Hybrid Methods

Native environments: We have shown in this thesis that high-density CLMS-based hybrid methods are able to probe protein structure in native environments. In section 9.3.2, we discussed the possibility of applying this methodology in the cell. However, this feature of CLMS-based hybrid methods might be also important for high-throughput structure determination, as we will discuss in the next paragraph.

High-throughput structure determination: Another advantage of CLMS-based hybrid methods is that data acquisition is relatively fast compared to other structure determination methods (about two weeks for a single protein). If these protocols can be further optimized, it is not unthinkable that CLMS data can be acquired in a few days. Even more importantly, the ability of CLMS methods to cope with complex environments of the protein sample opens up the possibility of high-throughput structure determination. Traditional structure determination requires highly purified sample. However, this purification step is labor and time consuming, and a bottleneck towards scaling up NMR and X-ray crystallography. By being able to cope with complex mixtures, CLMS might effectively overcome this limitation and enable CLMS-based protein structure determination in a high-throughput manner when coupled with automated, CLMS-driven structure prediction algorithms. The high-throughput determination of protein structure could have profound impact in life-science by an explosion of experimentally verified protein structure models.

9.5 Conclusion

This thesis devised algorithms for leveraging novel information sources for protein structure prediction. These information sources led to state-of-the-art contact and structure prediction algorithms that were rigorously validated in controlled and blind CASP experiments. In a proof-of-concept study, we presented a novel CLMS-driven hybrid method that is able to determine protein structure in natural environments.

We believe that we provided compelling evidence that these novel information sources are a promising addition to widely used information sources in structure prediction and might have long-term impact in the field. Furthermore, we strongly believe that further information sources are out there and are worth to be discovered.

The algorithms presented in this thesis might form the foundation for a novel class of search algorithms that tightly couple reasoning about constraints with search in high-dimensional spaces. Finally, this thesis laid down the groundwork for a experimental/ computational hybrid technique to address the ultimate challenge of observing protein architectures at their very place of action – inside the cell.

References

[1] Adhikari, A. N., Freed, K. F., and Sosnick, T. R. (2012). De novo prediction of protein folding pathways and structure using the principle of sequential stabilization. *Proc. Natl. Acad. Sci. U.S.A.*, 109(43):17442–17447.

[2] Adhikari, B., Bhattacharya, D., Cao, R., and Cheng, J. (2015). CONFOLD: Residue-residue contact-guided ab initio protein folding. *Proteins*, 83(8):1436–1449.

[3] Alexander, N., Al-Mestarihi, A., Bortolus, M., Mchaourab, H., and Meiler, J. (2008). De Novo High-Resolution Protein Structure Determination from Sparse Spin-Labeling EPR Data. *Structure*, 16(2):181–195.

[4] Šali, A. and Blundell, T. L. (1993). Comparative protein modelling by satisfaction of spatial restraints. *J. Mol. Biol.*, 234(3):779–815.

[5] Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., et al. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, 25(17):3389–3402.

[6] Amir, R. E., Van den Veyver, I. B., Wan, M., Tran, C. Q., Francke, U., et al. (1999). Rett syndrome is caused by mutations in X-linked MECP2, encoding methyl-CpG-binding protein 2. *Nat. Genet.*, 23:185–188.

[7] Aszódi, A. and Taylor, W. R. (1995). Estimating polypeptide alpha-carbon distances from multiple sequence alignments. *J. Math. Chem.*, 19(17):167–184.

[8] Baird, N. and Leidel, S. (2015). Structures to the people! *eLife*, 4:e09249.

[9] Bellman, R. (1962). *Adaptive Control Processes: A Guided Tour*. Princeton University Press, Princeton, NJ, USA.

[10] Belsom, A., Schneider, M., Fischer, L., Brock, O., and Rappsilber, J. (2015). Serum Albumin Domain Structures in Human Blood Serum by Mass Spectrometry and Computational Biology. *Mol. Cell. Proteomics*, in print: mcp.M115.048504.

[11] Berenger, F., Shrestha, R., Zhou, Y., Simoncini, D., and Zhang, K. Y. (2012). Durandal: fast exact clustering of protein decoys. *J. Comput. Chem.*, 33(4):471–474.

[12] Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., et al. (2000). The Protein Data Bank. *Nucleic Acids Res.*, 28(1):235–242.

[13] Bishop, C. (2007). *Pattern Recognition and Machine Learning*. Springer, New York, NY, USA.

[14] Björkholm, P., Daniluk, P., Kryshtafovych, A., Fidelis, K., Andersson, R., et al. (2009). Using multi-data hidden markov models trained on local neighborhoods of protein structure to predict residue-residue contacts. *Bioinformatics*, 25(10):1264–1270.

[15] Blencowe, A. and Hayes, W. (2005). Development and application of diazirines in biological and synthetic macromolecular systems. *Soft Matter*, 1(3):178–205.

[16] Blum, B., Jordan, M. I., and Baker, D. (2010). Feature space resampling for protein conformational search. *Proteins*, 78(6):1583–1593.

[17] Bonet, J. S. D., Isbell, C. L., Jr., and Viola, P. (1996). MIMIC: Finding Optima by Estimating Probability Densities. In *Procceedings of Advances in Neural Information Processing Systems 9*, NIPS '96, pages 424–430, Denver, CO, USA. The MIT Press.

[18] Bonomi, M., Pellarin, R., Kim, S. J., Russel, D., Sundin, B. A., et al. (2014). Determining protein complex structures based on a Bayesian model of in vivo Förster resonance energy transfer (FRET) data. *Mol. Cell. Proteomics*, 13(11):2812–2823.

[19] Borgwardt, K. and Kriegel, H.-P. (2005). Shortest-path kernels on graphs. In *Proceedings of the Fifth IEEE International Conference on Data Mining*, ICDM'05, pages 74–81, Houston, TX, USA. IEEE Computer Society.

[20] Borgwardt, K. M., Ong, C. S., Schönauer, S., Vishwanathan, S., Smola, A. J., et al. (2005). Protein function prediction via graph kernels. *Bioinformatics*, 21(Suppl 1):i47–i56.

[21] Boutet, S., Lomb, L., Williams, G. J., Barends, T. R. M., Aquila, A., et al. (2012). High-Resolution Protein Structure Determination by Serial Femtosecond Crystallography. *Science*, 337(6092):362–364.

[22] Boyan, J. A. (1998). *Learning Evaluation Functions for Global Optimization*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA.

[23] Bradley, P. and Baker, D. (2006). Improved beta-protein structure prediction by multilevel optimization of nonlocal strand pairings and local backbone conformation. *Proteins*, 65(4):922–929.

[24] Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5–32.

[25] Bremermann, H. (1962). Optimization through evolution and recombination. In *Proceedings of the Conference on Self-Organizing Systems, 1962*, pages 93–106, Washington, D.C. Spartan Books.

[26] Brunette, T. and Brock, O. (2008). Guiding Conformation Space Search with an All-Atom Energy Potential. *Proteins*, 73(4):958–972.

[27] Brünger, A., Adams, P., Clore, G., DeLano, W., Gros, P., et al. (1998). Crystallography & NMR system: A new software suite for macromolecular structure determination. *Acta Crystallogr. Sect. D-Biol. Crystallogr.*, 54:905–921.

[28] Buchan, D., Minneci, F., Nugent, T. C. O., Bryson, K., and Jones, D. T. (2013). Scalable web services for the PSIPRED Protein Analysis Workbench. *Nucleic Acids Res.*, 41(W1):W349–W357.

[29] Bullmore, E. and Sporns, O. (2009). Complex brain networks: graph theoretical analysis of structural and functional systems. *Nat. Rev. Neurosci.*, 10:186–198.

[30] Burges, C. J. C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Data Min. Knowl. Discov.*, 2(2):121–167.

[31] Burt, R. S., Kilduff, M., and Tasselli, S. (2013). Social Network Analysis: Foundations and Frontiers on Advantage. *Annu. Rev. Psychol.*, 64:527–547.

[32] Carpenter, E. P., Beis, K., Cameron, A. D., and Iwata, S. (2008). Overcoming the challenges of membrane protein crystallography. *Curr. Opin. Struct. Biol.*, 18(5):581–586.

[33] CASP Committee (2014). CASP11 Residue-residue Contact Prediction Results Page. http://www.predictioncenter.org/casp11/rr_summary_results.cgi.

[34] Cavallo, L., Kleinjung, J., and Fraternali, F. (2003). POPS: a fast algorithm for solvent accessible surface areas at atomic and residue level. *Nucleic Acids Res.*, 31(13):3364–3366.

[35] Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

[36] Chen, Z. A., Jawhari, A., Fischer, L., Buchen, C., Tahir, S., et al. (2010). Architecture of the RNA polymerase II-TFIIF complex revealed by crosslinking and mass spectrometry. *EMBO J.*, 29(4):717–726.

[37] Cheng, J. and Baldi, P. (2007). Improved residue contact prediction using support vector machines and a large feature set. *BMC Bioinform.*, 8:113.

[38] Cheng, J., Li, J., Wang, Z., Eickholt, J., and Deng, X. (2012). The MULTICOM toolbox for protein structure prediction. *BMC Bioinform.*, 13:65.

[39] Cheng, J., Sweredoski, M. J., and Baldi, P. (2006). DOMpro: Protein Domain Prediction Using Profiles, Secondary Structure, Relative Solvent Accessibility, and Recursive Neural Networks. *Data Min. Knowl. Discov.*, 13(1):1–10.

[40] Coin, I., Katritch, V., Sun, T., Xiang, Z., Siu, F. Y., et al. (2013). Genetically Encoded Chemical Probes in Cells Reveal the Binding Path of Urocortin-I to CRF Class B GPCR. *Cell*, 155(6):1258–1269.

[41] Cortes, C. and Vapnik, V. (1995). Support-Vector Networks. *Mach. Learn.*, 20(3):273–297.

[42] de Juan, D., Pazos, F., and Valencia, A. (2013). Emerging methods in protein co-evolution. *Nat. Rev. Genet.*, 14:249–261.

[43] de Oliveira, S. H., Shi, J., and Deane, C. M. (2015). Building a Better Fragment Library for De Novo Protein Structure Prediction. *PLoS ONE*, 10(4):e0123998.

[44] DeBartolo, J., Colubri, A., Jha, A. K., Fitzgerald, J. E., Freed, K. F., et al. (2009). Mimicking the folding pathway to improve homology-free protein structure prediction. *Proc. Natl. Acad. Sci. U.S.A.*, 106(10):3734–3739.

[45] Della Corte, D., Wildberg, A., and Schröder, G. F. (2015). Protein structure refinement with adaptively restrained homologous replicas. *Proteins*, in print.

[46] Di Lena, P., Nagata, K., and Baldi, P. (2012). Deep architectures for protein contact map prediction. *Bioinformatics*, 28(19):2449–2457.

[47] Do, C. B. and Batzoglou, S. (2008). What is the expectation maximization algorithm? *Nat. Biotechnol.*, 26:897–899.

[48] Domingos, P. (2012). A Few Useful Things to Know About Machine Learning. *Commun. ACM*, 55(10):78–87.

[49] Editorial in Nature (2014). A bittersweet celebration of crystallography.

[50] Eickholt, J. and Cheng, J. (2012). Predicting protein residue-residue contacts using deep networks and boosting. *Bioinformatics*, 28(23):3066–3072.

[51] Eickholt, J., Deng, X., and Cheng, J. (2011a). DoBo: Protein domain boundary prediction by integrating evolutionary signals and machine learning. *BMC Bioinform.*, 12:43.

[52] Eickholt, J., Wang, Z., and Cheng, J. (2011b). A conformation ensemble approach to protein residue-residue contact. *BMC Struct. Biol.*, 11:38.

[53] Ekeberg, M. (2013). Improved contact prediction in proteins: Using pseudolikelihoods to infer Potts models. *Phys. Rev. E*, 87(1):012707.

[54] Elias, J. E. and Gygi, S. P. (2007). Target-decoy search strategy for increased confidence in large-scale protein identifications by mass spectrometry. *Nat. Methods*, 4:207–214.

[55] Feig, M. and Mirjalili, V. (2015). Protein structure refinement via molecular-dynamics simulations: What works and what does not? *Proteins*, in print.

[56] Feizi, S., Marbach, D., Médard, M., and Kellis, M. (2013). Network deconvolution as a general method to distinguish direct dependencies in networks. *Nat. Biotechnol.*, 31:726–733.

[57] Fenn, J. B., Mann, M., Meng, C. K., Wong, S. F., and Whitehouse, C. M. (1989). Electrospray ionization for mass spectrometry of large biomolecules. *Science*, 246(4926):64–71.

[58] Fischer, J. D., Mayer, C. E., and Söding, J. (2008). Prediction of protein functional residues from sequence by probability density estimation. *Bioinformatics*, 24(5):613–620.

[59] Fischer, L., Chen, Z. A., and Rappsilber, J. (2013). Quantitative cross-linking/mass spectrometry using isotope-labelled cross-linkers. *J. Proteomics*, 88:120–128.

[60] Fischer, L. and Rappsilber, J. (2015). False discovery rate estimation in cross-linking/mass spectrometry. *Mol. Cell. Proteomics*, in revision.

[61] Förster, F., Webb, B., Krukenberg, K. A., Tsuruta, H., Agard, D. A., et al. (2008). Integration of Small-Angle X-Ray Scattering Data into Structural Modeling of Proteins and Their Assemblies. *J. Mol. Biol.*, 382(4):1089–1106.

[62] Frenkel-Morgenstern, M., Magid, R., Eyal, E., and Pietrokovski, S. (2007). Refining intra-protein contact prediction by graph analysis. *BMC Bioinform.*, 8(Suppl 5):S6.

[63] Frishman, D. and Argos, P. (1995). Knowledge-based protein secondary structure assignment. *Proteins*, 23(4):566–579.

[64] Gärtner, T., Flach, P., and Wrobel, S. (2003). On Graph Kernels: Hardness Results and Efficient Alternatives. In *Proceedings of the 16th Annual Conference on Learning Theory*, pages 129–143, Washington, DC, USA. Springer Berlin Heidelberg.

[65] Gibert, J., Valveny, E., and Bunke, H. (2012). Graph Embedding in Vector Spaces by Node Attribute Statistics. *Pattern Recogn.*, 45(9):3072–3083.

[66] Giese, S., Fischer, L., and Rappsilber, J. (2015). A Study into the CID Behavior of Cross-linked Peptides. *Mol. Cell. Proteomics*, in revision.

[67] Glickman, M. H. and Ciechanover, A. (2002). The Ubiquitin-Proteasome Proteolytic Pathway: Destruction for the Sake of Construction. *Physiol. Rev.*, 82(2):373–428.

[68] Glover, F. and Laguna, M. (1997). *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA.

[69] Göbel, U., Sander, C., Schneider, R., and Valencia, A. (1994). Correlated mutations and residue contacts in proteins. *Proteins*, 18(4):309–317.

[70] Gomes, A. F. and Gozzo, F. C. (2010). Chemical cross-linking with a diazirine photoactivatable cross-linker investigated by MALDI- and ESI-MS/MS. *J. Mass Spectrom.*, 45(8):892–899.

[71] Graña, O., Baker, D., MacCallum, R. M., Meiler, J., Punta, M., et al. (2005). CASP6 assessment of contact prediction. *Proteins*, 61(S7):214–224.

[72] Greene, L. H. (2012). Protein structure networks. *Brief. Func. Genom.*, 11(6):469–478.

[73] Gront, D., Kulp, D. W., Vernon, R. M., Strauss, C. E., and Baker, D. (2011). Generalized Fragment Picking in Rosetta: Design, Protocols and Applications. *PLoS ONE*, 6(8):e23294.

[74] Güntert, P. (2004). Automated NMR structure calculation with CYANA. *Methods Mol. Biol.*, 278:353–378.

[75] Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring Network Structure, Dynamics, and Function Using NetworkX. In *Proceedings of the 7th Python in Science Conference*, SciPy '11, pages 11–15, Pasadena, CA USA.

[76] Hauschild, M. and Pelikan, M. (2011). An introduction and survey of estimation of distribution algorithms. *Swarm Evol. Comput.*, 1(3):111–128.

[77] He, H. and Garcia, E. (2009). Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.*, 21(9):1263–1284.

[78] Hildebrand, A., Remmert, M., Biegert, A., and Söding, J. (2009). Fast and accurate automatic structure prediction with HHpred. *Proteins*, 77(S9):128–132.

[79] Hirst, S. J., Alexander, N., Mchaourab, H. S., and Meiler, J. (2011). RosettaEPR: An integrated tool for protein structure determination from sparse EPR data. *J. Struct. Biol.*, 173(3):506–514.

[80] Hofmann, T., Fischer, A. W., Meiler, J., and Kalkhof, S. (2015). Protein structure prediction guided by crosslinking restraints - A systematic evaluation of the impact of the crosslinking spacer length. *Methods*, in press.

[81] Holland, J. H. (1992). *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge, MA, USA.

[82] Huang, P.-S., Ban, Y.-E. A., Richter, F., Andre, I., Vernon, R., et al. (2011). RosettaRemodel: A Generalized Framework for Flexible Backbone Protein Design. *PLoS ONE*, 6(8):e24109.

[83] Janda, J.-O., Meier, A., and Merkl, R. (2013). CLIPS-4D: a classifier that distinguishes structurally and functionally important residue-positions based on sequence and 3D data. *Bioinformatics*, 29(23):3029–3035.

[84] Jie, B., Zhang, D., Wee, C.-Y., and Shen, D. (2014). Topological Graph Kernel on Multiple Thresholded Functional Connectivity Networks for Mild Cognitive Impairment Classification. *Hum. Brain Mapp.*, 35(7):2876–2897.

[85] Johnson, L. S., Eddy, S. R., and Portugaly, E. (2010). Hidden Markov model speed heuristic and iterative HMM search procedure. *BMC Bioinform.*, 11:431.

[86] Jones, D. T. (1999). Protein secondary structure prediction based on position-specific scoring matrices1. *J. Mol. Biol.*, 292(2):195–202.

[87] Jones, D. T. (2001). Predicting novel protein folds by using FRAGFOLD. *Proteins*, 45(S5):127–132.

[88] Jones, D. T., Buchan, D. W., Cozzetto, D., and Pontil, M. (2012). PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, 28(2):184–190.

[89] Jones, D. T., Singh, T., Kosciolek, T., and Tetchner, S. (2015). MetaPSICOV: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. *Bioinformatics*, 31(7):999–1006.

[90] Kabsch, W. and Sander, C. (1983). Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12):2577–2637.

[91] Kahraman, A., Herzog, F., Leitner, A., Rosenberger, G., Aebersold, R., et al. (2013). Cross-Link Guided Molecular Modeling with ROSETTA. *PLoS ONE*, 8(9):e73411.

[92] Kahraman, A., Malmström, L., and Aebersold, R. (2011). Xwalk: computing and visualizing distances in cross-linking experiments. *Bioinformatics*, 27(15):2163–2164.

[93] Kaján, L., Hopf, T. A., Kalaš, M., Marks, D. S., and Rost, B. (2014). FreeContact: fast and free software for protein contact prediction from residue co-evolution. *BMC Bioinform.*, 15:85.

[94] Kalev, I. and Habeck, M. (2011). HHfrag: HMM-based fragment detection using HHpred. *Bioinformatics*, 27(22):3110–3116.

[95] Kamburov, A., Grossmann, A., Herwig, R., and Stelzl, U. (2012). Cluster-based assessment of protein-protein interaction confidence. *BMC Bioinform.*, 13:262.

[96] Kamisetty, H., Ovchinnikov, S., and Baker, D. (2013). Assessing the utility of coevolution-based residue-residue contact predictions in a sequence- and structure-rich era. *Proc. Natl. Acad. Sci. U.S.A.*, 110(39):15674–15679.

[97] Kao, A., Randall, A., Yang, Y., Patel, V. R., Kandur, W., et al. (2012). Mapping the Structural Topology of the Yeast 19s Proteasomal Regulatory Particle Using Chemical Cross-linking and Probabilistic Modeling. *Mol. Cell. Proteomics*, 11(12):1566–1577.

[98] Karakaş, M., Woetzel, N., and Meiler, J. (2010). BCL::Contact-Low Confidence Fold Recognition Hits Boost Protein Contact Prediction and De Novo Structure Determination. *J. Comp. Biol.*, 17(2):153–168.

[99] Karakaş, M., Woetzel, N., Staritzbichler, R., Alexander, N., Weiner, B. E., et al. (2012). BCL::Fold - De Novo Prediction of Complex and Large Protein Topologies by Assembly of Secondary Structure Elements. *PLoS ONE*, 7(11):e49240.

[100] Karlebach, G. and Shamir, R. (2008). Modelling and analysis of gene regulatory networks. *Nat. Rev. Mol. Cell Biol.*, 9:770–780.

[101] Karplus, K. (2009). SAM-T08, HMM-based protein structure prediction. *Nucleic Acids Res.*, 37(suppl 2):W492–W497.

[102] Kinch, L., Yong Shi, S., Cong, Q., Cheng, H., Liao, Y., et al. (2011). CASP9 assessment of free modeling target predictions. *Proteins*, 79(S10):59–73.

[103] Kinch, L. N., Li, W., Monastyrskyy, B., Kryshtafovych, A., and Grishin, N. V. (2016). Evaluation of free modeling targets in CASP11 and ROLL. *Proteins*, in print.

[104] Kosciolek, T. and Jones, D. T. (2014). De Novo Structure Prediction of Globular Proteins Aided by Sequence Variation-Derived Contacts. *PLoS ONE*, 9(3):e92197.

[105] Kosciolek, T. and Jones, D. T. (2015). Accurate contact predictions using covariation techniques and machine learning. *Proteins*, in press.

[106] Kovacs, J. A., Yeager, M., and Abagyan, R. (2007). Computational Prediction of Atomic Structures of Helical Membrane Proteins Aided by EM Maps. *Biophys. J.*, 93(6):1950–1959.

[107] Krivov, G. G., Shapovalov, M. V., and Dunbrack, R. L. (2009). Improved prediction of protein side-chain conformations with SCWRL4. *Proteins*, 77(4):778–795.

[108] Kryshtafovych, A., Fidelis, K., and Moult, J. (2014). CASP10 results compared to those of previous CASP experiments. *Proteins*, 82(S2):164–174.

[109] Kukic, P., Mirabello, C., Tradigo, G., Walsh, I., Veltri, P., et al. (2014). Toward an accurate prediction of inter-residue distances in proteins using 2D recursive neural networks. *BMC Bioinform.*, 15:6.

[110] Kurgan, L., Cios, K., and Chen, K. (2008). SCPRED: Accurate prediction of protein structural class for sequences of twilight-zone similarity with predicting sequences. *BMC Bioinform.*, 9:226.

[111] Kwan, A. H., Mobli, M., Gooley, P. R., King, G. F., and Mackay, J. P. (2011). Macromolecular NMR spectroscopy for the non-spectroscopist. *FEBS J.*, 278(5):687–703.

[112] Lange, O. F. and Baker, D. (2012). Resolution-adapted recombination of structural features significantly improves sampling in restraint-guided structure calculation. *Proteins*, 80(3):884–895.

[113] Lange, O. F., Rossi, P., Sgourakis, N. G., Song, Y., Lee, H.-W., et al. (2012). Determination of solution structures of proteins up to 40 kDa using CS-Rosetta with sparse NMR data from deuterated samples. *Proc. Natl. Acad. Sci. U.S.A.*, 109(27):10873–10878.

[114] Langville, A. N. (2006). *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, Princeton, N.J, USA.

[115] Larrañaga, P. and Lozano, J. A. (2002). *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*, volume 2. Kluwer Academic Publishers, Boston, MA, USA.

[116] Lee, J., Scheraga, H. A., and Rackovsky, S. (1997). New optimization method for conformational energy calculations on polypeptides: Conformational space annealing. *J. Comput. Chem.*, 18(9):1222–1232.

[117] Levinthal, C. (1969). How to fold graciously. In *Mossbauer Spectroscopy in Biological Systems*, volume 67, pages 22–24, Monticello, IL, USA.

[118] Li, G., Semerci, M., Yener, B., and Zaki, M. J. (2012). Effective graph classification based on topological and label attributes. *Stat. Anal. Data Min.*, 5(4):265–283.

[119] Li, W., Zhang, Y., and Skolnick, J. (2004). Application of Sparse NMR Restraints to Large-Scale Protein Structure Prediction. *Biophys. J.*, 87(2):1241–1248.

[120] Li, Y., Fang, Y., and Fang, J. (2011). Predicting residue-residue contacts using random forest models. *Bioinformatics*, 27(24):3379–3384.

[121] Lindert, S., Staritzbichler, R., Wötzel, N., Karakaş, M., Stewart, P. L., et al. (2009). EM-Fold: De Novo Folding of α-Helical Proteins Guided by Intermediate-Resolution Electron Microscopy Density Maps. *Structure*, 17(7):990–1003.

[122] Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., and Watkins, C. (2002). Text Classification Using String Kernels. *J. Mach. Learn. Res.*, 2:419–444.

[123] Loquet, A., Sgourakis, N. G., Gupta, R., Giller, K., Riedel, D., et al. (2012). Atomic model of the type III secretion system needle. *Nature*, 486:276–279.

[124] Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, 60(2):91–110.

[125] Mabrouk, M., Putz, I., Werner, T., Schneider, M., Neeb, M., et al. (2015a). RBO Aleph: leveraging novel information sources for protein structure prediction. *Nucleic Acids Res.*, 43(W1):W343–W348.

[126] Mabrouk, M., Werner, T., Schneider, M., Putz, I., and Brock, O. (2015b). Analysis of Free Modeling Predictions by RBO Aleph in CASP11. *Proteins*, in press.

[127] MacCallum, J. L., Perez, A., and Dill, K. A. (2015). Determining protein structures by combining semireliable data with atomistic physical models by Bayesian inference. *Proc. Natl. Acad. Sci. U.S.A.*, 112(22):6985–6990.

[128] Maiolica, A., Cittaro, D., Borsotti, D., Sennels, L., Ciferri, C., et al. (2007). Structural Analysis of Multiprotein Complexes by Cross-linking, Mass Spectrometry, and Database Searching. *Mol. Cell. Proteomics*, 6(12):2200–2211.

[129] Mao, W., Kaya, C., Dutta, A., Horovitz, A., and Bahar, I. (2015). Comparative study of the effectiveness and limitations of current methods for detecting sequence coevolution. *Bioinformatics*, 31(12):1929–1937.

[130] Mariani, V., Kiefer, F., Schmidt, T., Haas, J., and Schwede, T. (2011). Assessment of template based protein structure predictions in CASP9. *Proteins*, 79(S10):37–58.

[131] Marks, D. S., Colwell, L. J., Sheridan, R., Hopf, T. A., Pagnani, A., et al. (2011). Protein 3D Structure Computed from Evolutionary Sequence Variation. *PLoS ONE*, 6(12):e28766.

[132] Marks, D. S., Hopf, T. A., and Sander, C. (2012). Protein structure prediction from sequence variation. *Nat. Biotechnol.*, 30:1072–1080.

[133] McGuffin, L. J. (2008). The ModFOLD server for the quality assessment of protein structural models. *Bioinformatics*, 24(4):586–587.

[134] Meier, A. and Söding, J. (2015). Automatic Prediction of Protein 3D Structures by Probabilistic Multi-template Homology Modeling. *PLoS Comput. Biol.*, 11(10):e1004343.

[135] Meiler, J. and Baker, D. (2003a). Coupled prediction of protein secondary and tertiary structure. *Proc. Natl. Acad. Sci. U.S.A.*, 100(21):12105–12110.

[136] Meiler, J. and Baker, D. (2003b). Rapid protein fold determination using unassigned NMR data. *Proc. Natl. Acad. Sci. U.S.A.*, 100(26):15404–15409.

[137] Merkley, E. D., Rysavy, S., Kahraman, A., Hafen, R. P., Daggett, V., et al. (2014). Distance restraints from crosslinking mass spectrometry: Mining a molecular dynamics simulation database to evaluate lysine-lysine distances. *Protein Sci.*, 23(6):747–759.

[138] Michel, M., Hayat, S., Skwark, M. J., Sander, C., Marks, D. S., et al. (2014). PconsFold: improved contact predictions improve protein models. *Bioinformatics*, 30(17):i482–i488.

[139] Mitchell, T. (1997). *Machine Learning*. McGraw-Hill, Inc, New York, NY, USA.

[140] Mitchell Wells, J. and McLuckey, S. A. (2005). Collision Induced Dissociation (CID) of Peptides and Proteins. *Meth. Enzymol*, 402:148–185.

[141] Monastyrskyy, B., D'Andrea, D., Fidelis, K., Tramontano, A., and Kryshtafovych, A. (2014). Evaluation of residue-residue contact prediction in CASP10. *Proteins*, 82(S2):138–153.

[142] Monastyrskyy, B., D'Andrea, D., Fidelis, K., Tramontano, A., and Kryshtafovych, A. (2015). New encouraging developments in contact prediction: Assessment of the CASP11 results. *Proteins*, in print.

[143] Monastyrskyy, B., Fidelis, K., Tramontano, A., and Kryshtafovych, A. (2011). Evaluation of residue-residue contact predictions in CASP9. *Proteins*, 79(S10):119–125.

[144] Moult, J., Fidelis, K., Kryshtafovych, A., Rost, B., Hubbard, T., et al. (2007). Critical assessment of methods of protein structure prediction-Round VII. *Proteins*, 69(S8):3–9.

[145] Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T., and Tramontano, A. (2014). Critical assessment of methods of protein structure prediction (CASP)-round x. *Proteins*, 82(S2):1–6.

[146] Moult, J., Fidelis, K., Zemla, A., and Hubbard, T. (2003). Critical assessment of methods of protein structure prediction (CASP)-round V. *Proteins*, 53(S6):334–339.

[147] Moult, J., Hubbard, T., Fidelis, K., and Pedersen, J. T. (1999). Critical assessment of methods of protein structure prediction (CASP): Round III. *Proteins*, 37(S3):2–6.

[148] Nanias, M., Chinchio, M., Ołdziej, S., Czaplewski, C., and Scheraga, H. A. (2005). Protein structure prediction with the UNRES force-field using Replica-Exchange Monte Carlo-with-Minimization; Comparison with MCM, CSA, and CFMC. *J. Comput. Chem.*, 26(14):1472–1486.

[149] Neumann, M., Patricia, N., Garnett, R., and Kersting, K. (2012). Efficient Graph Kernels by Randomization. In *Proceedings of the 2012 European conference on Machine Learning and Knowledge Discovery in Databases*, ECML PKDD '12, pages 378–393, Bristol, UK. Springer Berlin Heidelberg.

[150] Nugent, T., Cozzetto, D., and Jones, D. T. (2014). Evaluation of predictions in the CASP10 model refinement category. *Proteins*, 82 Suppl 2:98–111.

[151] Oostenbrink, C., Villa, A., Mark, A. E., and Van Gunsteren, W. F. (2004). A biomolecular force field based on the free enthalpy of hydration and solvation: The GROMOS force-field parameter sets 53A5 and 53A6. *J. Comput. Chem.*, 25(13):1656–1676.

[152] Ovchinnikov, S., Kim, D. E., Wang, R. Y.-R., Liu, Y., DiMaio, F., et al. (2015). Improved de novo structure prediction in CASP11 by incorporating Co-evolution information into rosetta. *Proteins*, in print.

[153] Ozkan, S. B., Wu, G. A., Chodera, J. D., and Dill, K. A. (2007). Protein folding by zipping and assembly. *Proc. Natl. Acad. Sci. U.S.A.*, 104(29):11987–11992.

[154] Page, L., Page, L., Brin, S., Motwani, R., and Winograd, T. (1998). The PageRank Citation Ranking: Bringing Order to the Web. *Technical Report, Stanford Digital Libraries*.

[155] Park, H., DiMaio, F., and Baker, D. (2015). CASP11 refinement experiments with ROSETTA. *Proteins*, in print.

[156] Park, H. W. and Thelwall, M. (2003). Hyperlink Analyses of the World Wide Web: A Review. *J. Comput-Mediat. Comm.*, 8(4):0.

[157] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., et al. (2011). Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830.

[158] Peng, J. and Xu, J. (2011). RaptorX: Exploiting structure information for protein alignment by statistical inference. *Proteins*, 79(S10):161–171.

[159] Perdigão, N., Heinrich, J., Stolte, C., Sabir, K. S., Buckley, M. J., et al. (2015). Unexpected features of the dark proteome. *Proc. Natl. Acad. Sci. U.S.A.*, 112(52):15898–15903.

[160] Pietal, M. J., Bujnicki, J. M., and Kozlowski, L. P. (2015). GDFuzz3D: a method for protein 3D structure reconstruction from contact maps, based on a non-Euclidean distance function. *Bioinformatics*, 31(21):3499–3505.

[161] Platt, J. C. (1999). Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. In *Advances in Large Margin Classifiers*, pages 61–74, Cambridge, MA, USA. MIT Press.

[162] Politis, A., Stengel, F., Hall, Z., Hernández, H., Leitner, A., et al. (2014). A mass spectrometry-based hybrid method for structural modeling of protein complexes. *Nat. Methods*, 11:403–406.

[163] Ponder, J. W. and Case, D. A. (2003). Force fields for protein simulations. *Adv. Protein Chem.*, 66:27–85.

[164] Punta, M. and Rost, B. (2005). PROFcon: novel prediction of long-range contacts. *Bioinformatics*, 21(13):2960–2968.

[165] Ramachandran, G. N., Ramakrishnan, C., and Sasisekharan, V. (1963). Stereochemistry of polypeptide chain configurations. *J. Mol. Biol.*, 7:95–99.

[166] Raman, S., Lange, O. F., Rossi, P., Tyka, M., Wang, X., et al. (2010). NMR Structure Determination for Larger Proteins Using Backbone-Only Data. *Science*, 327(5968):1014–1018.

[167] Rappsilber, J. (2011). The beginning of a beautiful friendship: Cross-linking/mass spectrometry and modelling of proteins and multi-protein complexes. *J. Struct. Biol.*, 173(3):530–540.

[168] Rappsilber, J., Siniossoglou, S., Hurt, E. C., and Mann, M. (2000). A Generic Strategy to Analyze the Spatial Organization of Multi-Protein Complexes by Cross-Linking and Mass Spectrometry. *Anal. Chem.*, 72(2):267–275.

[169] Remmert, M., Biegert, A., Hauser, A., and Söding, J. (2012). HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods*, 9:173–175.

[170] Rohl, C. A., Strauss, C. E., Misura, K. M., and Baker, D. (2004). Protein structure prediction using Rosetta. *Meth. Enzymol*, 383:66–93.

[171] Rost, B. (1999). Twilight zone of protein sequence alignments. *Protein Eng.*, 12(2):85–94.

[172] Roy, A., Kucukural, A., and Zhang, Y. (2010). I-TASSER: a unified platform for automated protein structure and function prediction. *Nat. Protoc.*, 5:725–738.

[173] Sakakibara, D., Sasaki, A., Ikeya, T., Hamatsu, J., Hanashima, T., et al. (2009). Protein structure determination in living cells by in-cell NMR spectroscopy. *Nature*, 458:102–105.

[174] Sali, A., Berman, H. M., Schwede, T., Trewhella, J., Kleywegt, G., et al. (2015). Outcome of the First wwPDB Hybrid/Integrative Methods Task Force Workshop. *Structure*, 23(7):1156–1167.

[175] Samish, I., Bourne, P. E., and Najmanovich, R. J. (2014). Achievements and challenges in structural bioinformatics and computational biophysics. *Bioinformatics*, 31(1):146–150.

[176] Samudrala, R., Xia, Y., Huang, E., and Levitt, M. (1999). Ab initio protein structure prediction using a combined hierarchical approach. *Proteins*, S3:194–198.

[177] Sathyapriya, R., Duarte, J. M., Stehr, H., Filippis, I., and Lappe, M. (2009). Defining an Essence of Structure Determining Residue Contacts in Proteins. *PLoS Comput. Biol.*, 5(2):e1000584.

[178] Schneider, M. and Brock, O. (2014). Combining Physicochemical and Evolutionary Information for Protein Contact Prediction. *PLoS ONE*, 9(10):e108438.

[179] Schwartz, M. (1986). *Telecommunication Networks: Protocols, Modeling and Analysis*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[180] Schwieters, C. D., Kuszewski, J. J., Tjandra, N., and Clore, G. M. (2003). The Xplor-NIH NMR molecular structure determination package. *J. Magn. Reson.*, 160(1):65–73.

[181] Seemayer, S., Gruber, M., and Söding, J. (2014). CCMpred—fast and precise prediction of protein residue-residue contacts from correlated mutations. *Bioinformatics*, 30(21):3128–3130.

[182] Shapovalov, M. V. and Dunbrack Jr., R. L. (2011). A Smoothed Backbone-Dependent Rotamer Library for Proteins Derived from Adaptive Kernel Density Estimates and Regressions. *Structure*, 19(6):844–858.

[183] Shell, M. S., Ozkan, S. B., Voelz, V., Wu, G. A., and Dill, K. A. (2009). Blind Test of Physics-Based Prediction of Protein Structures. *Biophys. J.*, 96(3):917–924.

[184] Shervashidze, N., Schweitzer, P., Leeuwen, E. J. v., Mehlhorn, K., and Borgwardt, K. M. (2011). Weisfeiler-Lehman Graph Kernels. *J. Mach. Learn. Res.*, 12:2539–2561.

[185] Shmygelska, A. and Levitt, M. (2009). Generalized ensemble methods for de novo structure prediction. *Proc. Natl. Acad. Sci. U.S.A.*, 106(5):1415–1420.

[186] Shrestha, R. and Zhang, K. Y. (2014). Improving fragment quality for de novo structure prediction. *Proteins*, 82(9):2240–2252.

[187] Sieradzan, A. K., Krupa, P., Scheraga, H. A., Liwo, A., and Czaplewski, C. (2015). Physics-Based Potentials for the Coupling between Backbone- and Side-Chain-Local Conformational States in the United Residue (UNRES) Force Field for Protein Simulations. *J. Chem. Theory Comput.*, 11(2):817–831.

[188] Sim, J., Kim, S.-Y., and Lee, J. (2005). PPRODO: Prediction of protein domain boundaries using neural networks. *Proteins*, 59(3):627–632.

[189] Simoncini, D., Berenger, F., Shrestha, R., and Zhang, K. Y. (2012). A Probabilistic Fragment-Based Protein Structure Prediction Algorithm. *PLoS ONE*, 7(7):e38799.

[190] Simons, K. T., Kooperberg, C., Huang, E., and Baker, D. (1997). Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring functions. *J. Mol. Biol.*, 268(1):209–225.

[191] Singh, P., Nakatani, E., Goodlett, D. R., and Catalano, C. E. (2013). A Pseudo-Atomic Model for the Capsid Shell of Bacteriophage Lambda Using Chemical Cross-Linking/Mass Spectrometry and Molecular Modeling. *J. Mol. Biol.*, 425(18):3378–3388.

[192] Sinz, A. (2006). Chemical cross-linking and mass spectrometry to map three-dimensional protein structures and protein-protein interactions. *Mass Spectrom. Rev.*, 25(4):663–682.

[193] Sippl, M. J. (1993). Recognition of errors in three-dimensional structures of proteins. *Proteins*, 17(4):355–362.

[194] Skwark, M. J., Raimondi, D., Michel, M., and Elofsson, A. (2014). Improved Contact Predictions Using the Recognition of Protein Like Contact Patterns. *PLoS Comput. Biol.*, 10(11):e1003889.

[195] Söding, J. (2005). Protein homology detection by HMM-HMM comparison. *Bioinformatics*, 21(7):951–960.

[196] Song, Y., DiMaio, F., Wang, R. Y.-R., Kim, D., Miles, C., et al. (2013). High-Resolution Comparative Modeling with RosettaCM. *Structure*, 21(10):1735–1742.

[197] Spolar, R. S., Ha, J. H., and Record, M. T. (1989). Hydrophobic effect in protein folding and other noncovalent processes involving proteins. *Proc. Natl. Acad. Sci. U.S.A.*, 86(21):8382–8385.

[198] Sun, H.-P., Huang, Y., Wang, X.-F., Zhang, Y., and Shen, H.-B. (2015). Improving accuracy of protein contact prediction using balanced network deconvolution. *Proteins*, 83(3):485–496.

[199] Tai, C.-H., Bai, H., Taylor, T. J., and Lee, B. (2014). Assessment of template-free modeling in CASP10 and ROLL. *Proteins*, 82(S2):57–83.

[200] Taylor, W. R., Hamilton, R. S., and Sadowski, M. I. (2013). Prediction of contacts from correlated sequence substitutions. *Curr. Opin. Struct. Biol.*, 23(3):473–479.

[201] Tegge, A. N., Wang, Z., Eickholt, J., and Cheng, J. (2009). NNcon: improved protein contact map prediction using 2D-recursive neural networks. *Nucleic Acids Res.*, 37(suppl 2):W515–W518.

[202] The UniProt Consortium (2015). UniProt: a hub for protein information. *Nucleic Acids Res.*, 41(D1):D204–D21.

[203] Tyka, M. D., Jung, K., and Baker, D. (2012). Efficient sampling of protein conformational space using fast loop building and batch minimization on highly parallel computers. *J. Comput. Chem.*, 33(31):2483–2491.

[204] Valentini, G. and Dietterich, T. G. (2003). Low Bias Bagged Support Vector Machines. In *Proceedings of the 20th International Conference on Machine Learning*, ICML'03, pages 752–759, Washington D.C, USA. AAAI Press.

[205] Vassura, M., Margara, L., Lena, P. D., Medri, F., Fariselli, P., et al. (2008). FT-COMAR: fault tolerant three-dimensional structure reconstruction from protein contact maps. *Bioinformatics*, 24(10):1313–1315.

[206] Venkatakrishnan, A. J., Deupi, X., Lebon, G., Tate, C. G., Schertler, G. F., et al. (2013). Molecular signatures of G-protein-coupled receptors. *Nature*, 494:185–194.

[207] Vishwanathan, S. V. N., Schraudolph, N. N., Kondor, R., and Borgwardt, K. M. (2010). Graph Kernels. *J. Mach. Learn. Res.*, 11:1201–1242.

[208] Vullo, A., Walsh, I., and Pollastri, G. (2006). A two-stage approach for improved prediction of residue contact maps. *BMC Bioinform.*, 7:180.

[209] Wal, A. L. J. T. and Boschma, R. A. (2008). Applying social network analysis in economic geography: framing some key analytic issues. *Ann. Reg. Sci.*, 43(3):739–756.

[210] Walzthoeni, T., Leitner, A., Stengel, F., and Aebersold, R. (2013). Mass spectrometry supported determination of protein complex structure. *Curr. Opin. Struct. Biol.*, 23(2):252–260.

[211] Wang, G. and Dunbrack, Roland L, J. (2003). PISCES: a protein sequence culling server. *Bioinformatics*, 19(12):1589–1591.

[212] Wang, J., Burdzinski, G., Kubicki, J., Platz, M. S., Moss, R. A., et al. (2006). Ultrafast Spectroscopic Study of the Photochemistry and Photophysics of Arylhalodiazirines: Direct Observation of Carbene and Zwitterion Formation. *J. Am. Chem. Soc.*, 128(51):16446–16447.

[213] Wang, R. Y.-R., Han, Y., Krassovsky, K., Sheffler, W., Tyka, M., et al. (2011). Modeling Disordered Regions in Proteins Using Rosetta. *PLoS ONE*, 6(7):e22060.

[214] Wang, S., Ma, J., Peng, J., and Xu, J. (2013). Protein structure alignment beyond spatial proximity. *Sci. Rep.*, 3:1448.

[215] Wang, Z. and Xu, J. (2013). Predicting protein contact map using evolutionary and physical constraints by integer programming. *Bioinformatics*, 29(13):i266–i273.

[216] Ward, A. B., Sali, A., and Wilson, I. A. (2013). Integrative Structural Biology. *Science*, 339(6122):913–915.

[217] Ward, J. J., Sodhi, J. S., McGuffin, L. J., Buxton, B. F., and Jones, D. T. (2004). Prediction and Functional Analysis of Native Disorder in Proteins from the Three Kingdoms of Life. *J. Mol. Biol.*, 337(3):635–645.

[218] Wilkinson, J. H. (1988). *The Algebraic Eigenvalue Problem*. Oxford University Press, Inc, New York, NY, USA.

[219] Wilson, D. N. (2014). Ribosome-targeting antibiotics and mechanisms of bacterial resistance. *Nat. Rev. Microbiol.*, 12:35–48.

[220] Winter, C., Kristiansen, G., Kersting, S., Roy, J., Aust, D., et al. (2012). Google goes cancer: improving outcome prediction for cancer patients by network-based ranking of marker genes. *PLoS Comput. Biol.*, 8(5):e1002511.

[221] Wollacott, A. M., Zanghellini, A., Murphy, P., and Baker, D. (2007). Prediction of structures of multidomain proteins from structures of the individual domains. *Protein Sci.*, 16(2):165–175.

[222] World Health Organization (2012). The evolving threat of antimicrobial resistance - Options for action. http://www.who.int/patientsafety/implementation/amr/publication/en/.

[223] Wu, S., Skolnick, J., and Zhang, Y. (2007). Ab initio modeling of small proteins by iterative TASSER simulations. *BMC Biol.*, 5:17.

[224] Wu, S., Szilagyi, A., and Zhang, Y. (2011). Improving Protein Structure Prediction Using Multiple Sequence-Based Contact Predictions. *Structure*, 19(8):1182–1191.

[225] Wu, S. and Zhang, Y. (2007). LOMETS: A local meta-threading-server for protein structure prediction. *Nucleic Acids Res.*, 35(10):3375–3382.

[226] Wu, S. and Zhang, Y. (2008). A comprehensive assessment of sequence-based and template-based methods for protein contact prediction. *Bioinformatics*, 24(7):924–931.

[227] Xu, D. and Zhang, Y. (2012). Ab initio protein structure assembly using continuous structure fragments and optimized knowledge-based force field. *Proteins*, 80(7):1715–1735.

[228] Xue, Z., Xu, D., Wang, Y., and Zhang, Y. (2013). ThreaDom: extracting protein domain boundary information from multiple threading alignments. *Bioinformatics*, 29(13):i247–i256.

[229] Yang, J. and Zhang, Y. (2015). I-TASSER server: new development for protein structure and function predictions. *Nucleic Acids Res.*, 41(W1):W174–W181.

[230] Yang, Y., Faraggi, E., Zhao, H., and Zhou, Y. (2011). Improving protein fold recognition and template-based modeling by employing probabilistic-based matching between predicted one-dimensional structural properties of query and corresponding native properties of templates. *Bioinformatics*, 27(15):2076–2082.

[231] Young, M. M., Tang, N., Hempel, J. C., Oshiro, C. M., Taylor, E. W., et al. (2000). High throughput protein fold identification by using experimental constraints derived from intramolecular cross-links and mass spectrometry. *Proc. Natl. Acad. Sci. U.S.A.*, 97(11):5802–5806.

[232] Zadrozny, B. and Elkan, C. (2001). Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Proceedings of the 18th International Conference on Machine Learning*, ICML '01, pages 609–616, Williamstown, MA, USA. Morgan Kaufmann Publishers Inc.

[233] Zemla, A., Venclovas, C., Moult, J., and Fidelis, K. (2001). Processing and evaluation of predictions in CASP4. *Proteins*, 45(S5):13–21.

[234] Zemla, A., Venclovas, e., Moult, J., and Fidelis, K. (1999). Processing and analysis of CASP3 protein structure predictions. *Proteins*, 37(S3):22–29.

[235] Zhang, W., Yang, J., He, B., Walker, S. E., Zhang, H., et al. (2015). Integration of QUARK and I-TASSER for Ab Initio Protein Structure Prediction in CASP11. *Proteins*, in print.

[236] Zhang, Y. (2014). Interplay of I-TASSER and QUARK for template-based and ab initio protein structure prediction in CASP10. *Proteins*, 82(S2):175–187.

[237] Zhang, Y., Kolinski, A., and Skolnick, J. (2003). TOUCHSTONE II: A New Approach to Ab Initio Protein Structure Prediction. *Biophys. J.*, 85(2):1145–1164.

[238] Zhang, Y. and Skolnick, J. (2004). Scoring function for automated assessment of protein structure template quality. *Proteins*, 57(4):702–710.

[239] Zhang, Y. and Skolnick, J. (2005). The protein structure prediction problem could be solved using the current PDB library. *Proc. Natl. Acad. Sci. U.S.A.*, 102(4):1029–1034.

[240] Zhao, F. and Xu, J. (2012). A Position-Specific Distance-Dependent Statistical Potential for Protein Structure and Functional Study. *Structure*, 20(6):1118–1126.

[241] Zhou, H. and Skolnick, J. (2011). GOAP: A Generalized Orientation-Dependent, All-Atom Statistical Potential for Protein Structure Prediction. *Biophys. J.*, 101(8):2043–2052.

[242] Zhu, J., Zhu, Q., Shi, Y., and Liu, H. (2003). How well can we predict native contacts in proteins based on decoy structures and their energies? *Proteins*, 52(4):598–608.

[243] Zwanzig, R., Szabo, A., and Bagchi, B. (1992). Levinthal's paradox. *Proc. Natl. Acad. Sci. U.S.A.*, 89(1):20–22.

Appendix A

Evaluation Criteria

A.1 Evaluation of Spatial Constraints

In this thesis, we evaluate spatial constraints from contact prediction with EPC-map (chapter 5 and 8) and from high-density cross-linking/mass spectrometry experiments (chapter 8 and 7). The spatial constraints in this thesis have a lower (usually 1.5 Å) and upper (8 Å or 20 Å) distance bound between specific atoms of the involved residues. We now describe the evaluation criteria to assess the quality of these constraints.

A.1.1 Contact Definition

Two residues are in contact if their C_β-C_β atom distance (C_α for glycine) is smaller or equal 8 Å in the native structure of a protein. We also distinguish contacts by the sequence separation of the involved residues. Medium-range contacts (sequence separation $12-23$) mostly represent interactions between adjacent secondary structures. Long-range contacts (sequence separation > 23) represent interactions of secondary structures that are not adjacent in sequence. Long-range contacts indicate global properties of protein structure and are therefore more valuable for protein modeling [177]. Thus, we mostly focus on evaluating long-range contacts, especially when we present the results of EPC-map in chapter 5.

A.1.2 CLMS Definition

A sulfo-SDA cross-link between two residues does not provide a well defined spatial constraint, because the exact insertion atom of the photo-reactive side of the linker is unknown. In addition, the exact conformation of cross-linked side-chains is unknown. To incorporate the uncertainty of all of these factors, we define the upper distance bound of CLMS con-

straints as 20 Å between C_α atoms. Since, we do not know the exact reacted atom and current mass spectrometry methods cannot exactly pinpoint the reacted atom, we assume that the linker might insert into any atom of the residue. Thus, we always use C_α atoms to define the anchor atoms of the constraint. We use 20 Å as the upper distance bound because this is the maximum Euclidean distance between C_α atoms of amino acids with long side chains (lysine and arginine, approximately 7 Å) plus the linker distance of sulfo-SDA (3.9 Å). We also add some extra distance to account for conformational flexibility. Note that due to the uncertainty of the residue assignment we refrain from using amino acid specific upper distance bounds, which would result in tighter distance bounds for shorter cross-linked amino acids.

A.1.3 Evaluation Criteria for Spatial Constraints

We define a true positive (TP) spatial constraint if the distance in the experimental structure is within the lower and upper constraint distance bound. Otherwise, the spatial constraint is considered to be a false positive (FP).

Please note that this definition is not straightforward to apply for CLMS constraints. A CLMS constraint might be long distance (i.e. the residue–residue distance in the experimental structure is longer than 20 Å) but might still be a true positive cross-link in terms of spectrum match, database search, scoring. Such links might also arise from cross-linking conformational states other than the crystal structure in solution. Strictly speaking, the cross-link is a true positive in terms of the CLMS approach. Thus, the term *long-distance* CLMS constraint is more correct. However, in lieu of defining the evaluating criteria for contacts and CLMS constraints twice with the same formulas but different nomenclature, we will still use the true positive and false positive definition from above for CLMS constraints.

We typically consider the top scoring fraction of $L/10$, $L/5$ and $L/2$ etc. spatial constraints, with L being the length of the protein. We define the accuracy of the spatial constraints as:

$$Acc = \frac{TP}{TP+FP},$$

where TP are the true positives and FP the false positives of the top scoring constraints. Note that the machine learning community refers to this metric as the *precision* or *positive predicted value* when evaluating binary classifiers. However, accuracy and precision are used interchangeably in computational biology, especially when evaluating contact prediction algorithms.

Additionally, we define the coverage as:

$$Cov = \frac{TP_{frac}}{TP_{total}},$$

where TP_{frac} is the number of true positives in the top scoring fraction and TP_{total} is the number of the total true positive constraints of the protein structure.

We also evaluate the quality of spatial constraints by the receiver operator characteristic (ROC). A ROC curve plots the *true positive rate* (TPR) against the false positive rate (FPR) at several score thresholds. The TPR is defined as

$$TPR = \frac{TP}{TP+FN},$$

and the FPR as

$$FPR = \frac{TN}{TN+FP}.$$

For a perfect score function, all true positive spatial constraints would be at the top of the list, followed by all false positive spatial constraints. In such a case, the ROC curve would immediately rise to a TPR of 100 % and stay there for all FPR values. For a randomly sorted list, the curve would lie close to the diagonal. We also use the *Area Under ROC-curve* (AUROC) to evaluate spatial constraints. This area is proportional to the probability that a uniformly chosen positive example ranks higher than a uniformly chosen negative example.

A.2 Evaluation of Protein Structure Models

This section describes the evaluation criteria for assessing three-dimensional models of protein structure. Since this thesis is mainly concerned with *ab initio* structure prediction, we focus on evaluation criteria of backbone conformations. Unless otherwise stated, we apply the following evaluation criteria on the C_α on the compared protein structures. All measures are computed after superposition of the model structure on the experimental structure.

A.2.1 Root-Mean-Square Deviation

The most widely used measure to assess the backbone quality of a model is the C_α-root-mean-square deviation (C_α-RMSD). The RMSD quantifies the deviation of the C_α coordinates with the following formula:

$$\mathrm{RMSD}(\mathbf{x_1}, \mathbf{x_2}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|x_1^i - x_2^i\|^2}.$$

Here, $\mathbf{x_1}$ and $\mathbf{x_2}$ are the sets of C_α-atom coordinates in the two structures, x^i the coordinate of the ith C_α-atom, and n the number of compared C_α-atoms.

The RMSD is sensitive to small changes in the model. However, this also leads to the sensitivity of the RMSD to outliers. Because the RMSD treats the deviation of all atoms equally, a few atoms with high deviation can lead to high RMSD values, even if the overall fold is correct. The following criteria have been developed to address this issue.

A.2.2 Global Distance Test

Since the RMSD measure is sensitive to large, local deviations of individual atoms, Zemla et al. [233] introduced the global distance test/total score (GDT_TS) measure to compare predicted and native protein structures. The global distance test measures the number of residues that deviate at most by a specified distance value [234]. The GDT_TS measure is the mean fraction of superimposable residues at 1, 2, 4, and 8 Å distance cutoffs [233]. This is a more robust measure of overall fold correctness, even if some regions of the protein are incorrectly modeled and therefore have high RMSD values.

A.2.3 Template-Modeling Score

The RMSD and GDT_TS measures usually assume that the predicted and experimental structure have the same length. However, this measure is less useful when comparing structures of different lengths, such as as partial alignments of templates onto the target structure. For example, an alignment with a high GDT_TS value is not useful to model the full-length target structure when it only covers a small part of the sequence. Zhang and Skolnick [238] devised the template-modeling score (TM-score), which normalizes modeling errors by a protein-size dependent factor to eliminate the bias to the target protein length. The TM-score is defined as

$$\text{TM-score} = \max \left[\frac{1}{L_{\text{target}}} \sum_{i}^{L_{\text{aligned}}} \frac{1}{1 + \left(\frac{d_i}{d_0(L_{\text{target}})} \right)^2} \right],$$

where L_{target} is the length of the protein target and L_{aligned} the length of the aligned region. The distance between the ith pair of atoms is given by d_i. $d_0(L_{\text{target}})$ is a length-dependent normalization factor that is given by: $d_0(L_{\text{target}}) = 1.24 \sqrt[3]{L_{\text{target}} - 15} - 1.8$.

Appendix B

Features in EPC-map

The detailed description of the features in EPC-map has been previously published in the supporting information of the following publication:

Schneider, M. and Brock, O. (2014). Combining Physicochemical and Evolutionary Information for Protein Contact Prediction. PLoS ONE, 9(10):e108438.

B.1 Graphs for Modeling Physicochemical Context

The main contribution of EPC-map is the prediction of contacts with physicochemical information. EPC-map extracts the physicochemical information from the local contact context of protein, which we model as a graph (see Figure 5.2). In these graphs, nodes represent residues and edges the contacts in the analyzed decoys. In this section, we describe the node and edge labels in contact graphs. In the next section, we present a detailed description of the (physicochemical) features in EPC-map.

B.1.1 Node Labels

Chemical type: We classify chemical properties of a residue into four categories: Non-polar, polar, acidic and basic.

Secondary structure: We classify the secondary structure into helix, sheet, turn or coil.

Solvent accessibility: We compute the residue-wise solvent accessibility with POPS [34]. We label residues with a relative solvent accessibility $> 25\%$ as exposed and buried otherwise.

Free solvation energy: The POPS free solvation energy of the residue.

Table B.1 Summary of node labels. Table source: Schneider and Brock [178].

Node label	Possible labels
Chemical type	Non-polar, polar, acidic, basic
Secondary structure	Helix, sheet, turn, coil
Solvent accessibility	Buried, exposed
Free solvation energy	Continuous value
Secondary structure length	Discrete value
Secondary structure 3D length	Continuous value
Secondary structure buried	Continuous value
Secondary structure exposed	Continuous value
Hydrogen bonding	Donor, acceptor, not involved
Distance to the centroid	Continuous value
Sequence conservation	Continuous value
Sequence neighborhood conservation	Continuous value

Secondary structure length: This label is the length of the secondary structure element in number of amino acids that contains this residue.

Secondary structure 3D length: The distance (in Å) between the C_α atoms of the first and last residue of the secondary structure element that contains.

Secondary structure buried: The number of buried amino acids in the secondary structure element that contains this residue.

Secondary structure exposed: The mean number of exposed amino acids (averaged over the decoys) in the secondary structure element that contains this residue.

Hydrogen bonding: Possible values are donor, acceptor or not involved in hydrogen bonding.

Distance to the centroid: The distance of the C_α atom of the residue to the structure centroid in Å.

Sequence conservation: Conservation measure of the residue in the multiple-sequence alignment, as introduced in [58, 83].

Sequence neighborhood conservation: The sequence conservation in the neighborhood $i-3, i-2\, i-1, i+1, i+2\, i+3$ of residue i as in [58, 83].

B.1.2 Edge Labels

Table B.2 Summary of edge labels. Table source: Schneider and Brock [178].

Edge label	Possible labels
Contact potential	Continuous value
3D distance	Continuous value
Sequence separation	Discrete value
Mutual information	Continuous value

Contact potential: Li et al. [120] devised this potential by analyzing contacts between particular types of amino acids (for example, ASP-GLU) in high-resolution crystal structures.

3D distance: C_β atom distance of the contacting residues (C_α for glycine).

Sequence separation: We measure the sequence separation between contacting amino acids in number of amino acids.

Mutual information: The mutual information between i and j in the multiple-sequence alignment.

B.2 Features in EPC-map

We designed a number of features to capture physicochemical context properties of contacting residues. Each feature maps a specific property to a number of binary and/or continuously valued inputs. We concatenate all inputs to generate the final input vector.

We group the features into seven categories: Pairwise, graph topology, graph spectrum, single node, node label statistics, edge label statistics and whole protein features. In this section, we give a detailed description of each feature that is not self-explanatory. The following tables list all features used in EPC-map.

Table B.3 Pairwise features between contacting residues. Table source: Schneider and Brock [178].

Feature	Description	Number of inputs
Chemical type	Chemical type of the contacting amino acids: non-polar, polar, acidic, basic	10[a]
Secondary structure	Secondary structure of the contacting amino acids: helix, sheet, turn, coil	10[a]
Solvent accessibility	Solvent accessibility of the contacting amino acids: exposed, buried	3[a]
Hydrogen bonding	Hydrogen bonding state of the contacting amino acids: donor, acceptor	2[a]
Sequence separation	Sequence separation encoded in 17 bins	17[a]
Sequence separation from N/C-terminus	Distance in amino acids between i and N-terminus; j and C-terminus	2
Contact potential	Contact potential from Li et al. [120]	1
Distance	3D distance between i and j	1
Mutual information	Sequence mutual information	1
Ensemble distance	Mean distance and standard deviation of i and j in ensemble if $d_{ij} \leq 12$ Å	2
Total inputs		49

[a] Binary inputs

B.2.1 Pairwise Residue Features

Pairwise features capture any direct physicochemical or structural properties of the contacting residue pairs. We encode categorical features of the contacting residues i and j by a series of binary inputs. If a property can be described by two states s_1 and s_2 this results in a bit vector $[1,0,0]^T$ if both residues are in s_1, $[0,1,0]^T$ if one residue is in s_1 and the other in s_2, and $[0,0,1]^T$ if both residues are in state s_2. Table B.3 provides an overview of pairwise features.

Chemical type: Since we classify the chemical type of residues into four classes(non-polar, polar, acidic and basic), this results in 10 combinations for a contact pair [37] (10 inputs).

Secondary structure: All combinations of the secondary structure state as determined by STRIDE [63] (10 inputs).

Solvent accessibility: Solvent accessibility states of the contacting residues (see section B.1.1) (3 inputs).

Hydrogen bonding: Binary feature that assesses whether the contacting residues as a donor or acceptor (2 inputs).

Sequence separation: We encode the sequence separation of the contacting residues by 17 binary inputs (12-13, 14-15, 16-17, 18-19, 20-21, 22-23, 24-28, 29-32, 33-36, 37-40, 41-44, 45-48, 49-52, 53-57, 58-62, 63-67, $<$68) (17 inputs).

Sequence separation from N/C-terminus: The sequence separation to the N- or C-terminus, measured from the residue closest to the respective terminus. (2 inputs).

Mutual information: Sequence-alignment mutual information between residues i and j (1 input).

Ensemble distance: Average and the standard deviation of the 3D-distance of the contact in all decoys where this distance is $\leq$12 Å (2 inputs).

B.2.2 Graph Features

The main hypothesis of EPC-map is that native and non-native contacts differ in their immediate neighborhood. Thus, measuring the similarity of contact graphs that capture this neighborhood should discriminate native and non-native contacts. We extract graph features from the contact graphs to accomplish this task.

Graph features capture topological properties or node/edge label statistics. Li et al. [118] showed that this approach is competitive with other graph kernel approaches in graph classification. We also design a number of features that are specific to the domain of contact prediction. The features translate graph properties into a vectorial representation. The vectors can be compared with standard kernel functions. We use NetworkX to carry out most of the graph-based calculations [75].

We extract each graph feature separately for the immediate and shared neighborhood graphs (see Figure 5.2). Thus, each graph feature is present twice in the final feature vector.

Graph Topology

Graph topology features capture topological properties of the contact graphs (Table B.4).

Table B.4 Graph topology features. Table source: Schneider and Brock [178].

Feature	Description	Number of inputs
Number of nodes	Number of nodes in the graph	1
Number of edges	Number of edges in the graph	1
Average degree centrality	See text	1
Average closeness centrality	See text	1
Average betweenness centrality	See text	1
Average eccentricity	See text	1
Graph radius	See text	1
Graph diameter	See text	1
Number of end points	See text	1
Average clustering coefficient	See text	1
Total inputs		10

Average degree centrality: For a node in a graph, the degree centrality is the fraction of nodes that it is connected to. This feature captures the average degree centrality over all nodes. We view this as a measure of packing density of the contact network. Tightly packed regions have a larger average degree centrality (1 input).

Average closeness centrality: For a node in a graph, The average closeness centrality is the reciprocal average path length to all other nodes. We use the average closeness centrality as a feature. This is another measure of packing density of the residues in the contact graph (1 input).

Average betweenness centrality: For a node in a graph, the betweenness centrality is the number of shortest paths from all shortest paths in the graph that pass through the node. Again, we take the average over all nodes in this feature. This measures the number of contact-mediated short-cuts in the contact graph. This relates to the loss in entropy by the formation of the contact network (1 input).

Average eccentricity: For a node in a graph, eccentricity is the length of the longest of all all-pair shortest paths that pass trough that node (1 input).

Graph radius: The graph radius is the smallest eccentricity value of all nodes (1 input).

Graph diameter: The graph diameter is the largest eccentricity value of all nodes (1 input).

Number of end points: Number of nodes in the graph with an degree of one (1 input).

Average clustering coefficient: The clustering coefficient is the ratio of actual edges between the neighbors of a node and the number of possible edges between the node and the neighbors (1 input).

Graph Spectrum

Table B.5 Graph spectrum features. Table source: Schneider and Brock [178].

Feature	Description	Number of inputs
Largest eigenvalue	Largest eigenvalue	1
Second largest eigenvalue	Second largest eigenvalue	1
Number of different eigenvalues	Number of different eigenvalues	1
Sum of eigenvalues	Trace of the adjacency matrix	1
Energy	Sum of squared eigenvalues	1
Total inputs		5

We extract graph spectrum features from the adjacency matrix of the graph (Table B.5).

Number of different eigenvalues: The number of different eigenvalues (1 input).

Sum of eigenvalues: The sum of all eigenvalues (1 input).

Energy: The energy of the adjacency matrix is the sum of the squared eigenvalues (1 input).

Single Node Features

Single node features capture topological characteristics of nodes i and j. We calculate these features for separately for i and j (Table B.6).

Node Label Statistics

Node label statistics describe the distribution of node labels in the graph (Table B.7). For binary labels, we count the number of specific node labels. We discretize continuous values

Table B.6 Single node features. Table source: Schneider and Brock [178].

Feature	Description	Number of inputs
Degree	Node degree in the graph	1
Closeness centrality	Reciprocal average path length from the node to all other nodes	1
Betweenness centrality	Average number of shortest paths that pass through the node	1
Sequence conservation	Conservation of residue position in multiple sequence alignment	1
Sequence neighborhood conservation	Conservation of neighboring residues in multiple-sequence alignment	1
Total inputs		5

into bins. The distribution is the number of counts of the continuous labels that falls into each bin.

We calculate the node label statistics separately for each type of label, such as secondary structure and solvent accessibility.

Chemical type: Counts of polar, non-polar, basic or acidic labels nodes, which measures chemical amino acid composition of the graph (4 inputs).

Secondary structure: Counts of nodes with helix, sheet, turn and coil labels (4 inputs).

Secondary structure length: Average lengths (in residues) of the secondary structure elements in the graph. We calculate this feature separately for each secondary structure type, resulting in 4 inputs. We follow the same procedure for all other secondary structure types (4 inputs).

Secondary structure 3D length: The average length of the secondary structure elements (4 inputs).

Secondary structure buried: Average number of buried residues in a specific type of secondary structure (helix, for example) (4 inputs).

Secondary structure exposed: Average number of exposed residues in a specific type of secondary structure (helix, for example) (4 inputs).

Table B.7 Node label statistics. Table source: Schneider and Brock [178].

Feature	Description	Number of inputs
Chemical type	Number of polar, non-polar, acidic, basic labels	4
Secondary structure	Number of nodes with helix, sheet, turn, coil labels	4
Secondary structure length	Average length of secondary structure element in amino acids	4
Secondary structure 3D length	Average 3D length of secondary structure element	4
Secondary structure buried	Average number of buried residues in secondary structure element	4
Secondary structure exposed	Average number of exposed residues in secondary structure element	4
Solvent accessibility	Number of exposed/buried nodes	2
Hydrogen bonding	Number of nodes that act as donor, acceptor or do not form hydrogen bonds	3
Average solvation energy	Average free solvation energy	1
Solvation energy distribution	4-bin distribution of free solvation energy	4
Label entropy	Entropy of the labels	3
Neighborhood impurity degree	Average number of neighbors with different labels	3
Distance to centroid	Average distance of nodes to the centroid	1
Sequence conservation	Average sequence conservation of nodes	1
Sequence neighborhood conservation	Average sequence neighborhood conservation of nodes	1
Total inputs		43

Solvent accessibility: Total number of nodes with the buried or exposed node label (2 inputs).

Hydrogen bonding: Total number of nodes that form hydrogen bonds as a donor, acceptor, or do not form any hydrogen bonds (3 inputs).

Average solvation energy: Average residue-wise solvation energy of all nodes in the graph (1 input).

Solvation energy distribution: The solvation energies in the graph, discretized into four bins (4 inputs).

Label entropy: We calculate the label entropy for chemical type, secondary structure and solvent accessibility labels (3 inputs).

Neighborhood impurity degree: For a node, this feature calculates the number of neighbors with different labels. We calculate the neighborhood impurity degree for chemical type, secondary structure and solvent accessibility labels (3 inputs).

Edge Label Statistics

Edge label statistics describe the distribution of edge labels in the graph (Table B.8).

Table B.8 Edge label statistics. Table source: Schneider and Brock [178].

Feature	Description	Number of inputs
Link impurity	Number of edges connecting two nodes with different labels	3
Mutual information distribution	5-bin distribution of mutual information	5
Cumulative mutual information	Cumulative mutual information over all edges	1
Contact potential	3-bin distribution of contact potential	3
Total inputs		12

Link impurity: The link impurity calculates the fraction of edges between neighboring nodes with different labels. We calculate the neighborhood impurity degree for chemical type, secondary structure and solvent accessibility labels (3 inputs).

Mutual information distribution: We calculate a sequence separation-dependent (5-bins) sequence mutual information distribution (adjacent, 2-6, 7-11, 12-23, $>$24)(5 inputs).

Contact potential: We use the potential of Li et al. [120] to calculate the contact potential distribution in the graph. We form bins over edges with low ($<$0.1), medium (0.1-0.3) or high ($>$0.3) contact potential. The feature then encodes the number of edges in each bin (3 inputs).

B.2.3 Whole Protein Features

This last group of features capture global features of the protein (Table B.9).

Table B.9 Whole protein features. Table source: Schneider and Brock [178].

Feature	Description	Number of inputs
Amino acid composition	Occurrence of each amino acid in the protein	20
Secondary structure composition	Occurrence of secondary structure in decoy	4
Length class	Binned length of the protein	5[a]
Total inputs		29

[a] Binary inputs

Amino acid composition: The length-normalized count of each amino acids (20 inputs).

Secondary structure composition: The number of distinct secondary structures (helix, sheet, turn, coil) in the decoy (4 inputs).

Length class: The length of the protein sequence in five bins ($<$60, 60-89, 90-119, 120-149, $>$150) (5 inputs).

All features form an input vector with length 228. Please note again that we calculate all graph-based features separately for the shared and immediate neighborhood graph. Thus, graph-based features are present twice in the final input vector.

Appendix C

Training and Test Set of EPC-map

This appendix lists the training and test sets of EPC-map in the following format: PBD_ID number_of_residues alignment_size

C.1 Training set of EPC-map (EPC-map_train)

1a6sA 87 12
1aa3A 63 1362
1abvA 105 2013
1ahoA 64 294
1bb8A 71 230
1beaA 116 249
1bhuA 102 1
1bm8A 99 393
1bo9A 73 1650
1brzA 54 1
1bw3A 125 2821
1bxyA 60 1480
1bz4A 144 346
1c1lA 135 1118
1c3yA 108 1208
1c5eA 95 42
1cdbA 105 932
1cfeA 135 3814
1d0dA 60 2
1d0qA 102 3116
1defA 147 2922
1dj7A 109 209
1dk8A 147 2119
1dm9A 104 13311
1dp7P 76 291
1dqcA 73 2136
1droA 122 6014
1dtdB 61 1
1ehxA 94 152
1eikA 77 308
1eiwA 111 945
1ej5A 107 694
1ej8A 140 1747
1ezjA 114 4
1f39A 101 6181
1f46A 139 322
1f6vA 91 54
1f7dA 118 3085
1f86A 115 787
1fhoA 119 916
1fybA 111 89
1g2hA 61 17455
1g7dA 106 202
1g7eA 122 14530
1gakA 137 5
1gh9A 71 68
1ghhA 81 199
1gp0A 133 790
1gu2A 124 4525
1gvjA 141 748
1gvpA 87 48
1h03P 125 4126
1h8bA 73 9473
1h8pA 88 569
1h9eA 56 33
1ha8A 51 1
1hdlA 55 1007
1hp8A 68 265
1hufA 123 6
1hx2A 60 1593
1hypA 74 286
1hywA 58 73
1hz6A 67 5
1i35A 95 69
1i4jA 110 1928
1i71A 83 1135
1ifrA 113 599
1ig6A 107 791
1iioA 84 67
1iqoA 88 22
1iw4A 55 1
1j0tA 78 121
1j2lA 68 1019
1j2mA 99 92
1j57A 143 102
1j5uA 127 387
1j7qA 86 15537
1j8bA 92 1616
1j9iA 68 15871
1jbiA 100 559
1jiwI 105 139
1jmvA 140 13473
1jo0A 97 984
1jr5A 90 20
1jr6A 138 27568
1jrmA 104 913
1jt8A 102 1479
1jvrA 137 9
1jxcA 68 16
1k5kA 87 1571
1k6kA 142 4031
1k8mA 87 25550
1kg1A 60 1
1kjsA 74 217
1kmxA 55 37
1kp6A 79 1
1kptA 105 53
1ksqA 75 289
1l6pA 121 879
1lniA 96 375
1lpvA 52 201
1lv3A 65 735
1lwbA 122 267
1m2dA 101 3809
1mc2A 122 883
1mknA 59 35
1mp1A 111 340
1n0qA 91 50056
1n6zA 105 29
1n91A 108 914
1nepA 130 648
1neqA 74 27900
1ng6A 148 2918
1nnvA 107 61
1nr3A 122 5004
1ny4A 71 175
1nz0A 109 1750
1o4wA 125 7437
1o8rA 94 64
1oa8A 128 344
1of9A 77 874
1oh1A 109 3
1ok0A 74 11
1oz9A 141 1924
1p4pA 140 43
1p57A 110 2850
1p9kA 79 13324
1pdoA 129 1993
1pfsA 78 30
1pqhA 119 812
1pqxA 91 667
1pulA 103 201
1pzwA 80 1583
1q2zA 120 185
1q9uA 128 948
1qj8A 148 4599
1qw1A 121 1176
1qzmA 94 1845
1r0rI 51 2387
1r7lA 103 34
1rykA 69 1504
1s3cA 138 3130
1s7zA 106 9
1sb0A 87 108
1sbxA 106 105
1seiA 130 1857
1sfpA 111 3564
1sg5A 86 163
1sgoA 139 267
1skzA 104 120
1snlA 99 14475
1srvA 145 1463
1ss6A 102 383
1t07A 81 313
1t0gA 109 2628
1t1dA 100 1648
1t1jA 119 911
1t23A 93 51
1t50A 58 4
1tdpA 111 70
1te4A 111 11295
1tfeA 142 2691
1tifA 74 1261
1tk7A 88 2310
1tleA 58 2023
1tm9A 137 2
1tnsA 76 339
1tqzA 133 167
1tuwA 106 126
1tuzA 118 1295
1u07A 90 4109
1u9lA 68 1441
1ucsA 64 951
1udkA 51 773
1ufyA 121 292
1ufzA 83 56
1ug7A 128 29
1ugiA 83 1
1ugjA 141 84
1uoyA 64 5

1ufyA 121 292
1ufzA 83 56
1ug7A 128 29
1ugiA 83 1
1ugjA 141 84
1uoyA 64 5
1utgA 70 176
1v30A 118 1702
1v32A 101 782
1v5rA 97 279
1v74A 107 25
1v74B 87 29
1v95A 130 56
1vbwA 68 527
1vccA 77 336
1vegA 83 2534
1vlkA 142 198
1vmhA 128 1474
1vr7A 120 681
1vyiA 111 18
1w2qA 127 461
1w6vA 120 605
1wgwA 99 2276
1wh0A 134 1774
1wh9A 92 1444
1whrA 124 763
1whzA 70 1652
1widA 117 1264
1wj7A 104 1661
1wjkA 100 21605
1wjpA 107 32163
1wjwA 112 5221
1wkaA 143 6170
1wktA 88 2
1wlxA 129 97
1wlzA 85 16484
1wn2A 118 692
1wouA 119 18504
1ws8A 104 1293
1wwjA 99 5195
1wwtA 88 4683
1x3aA 100 447
1x3bA 146 2906
1x3qA 57 1759
1x4pA 66 583
1x4tA 92 159
1x52A 124 961
1x53A 145 6224
1x65A 89 4176
1x9lA 149 1012
1xeeA 91 1
1xs0A 128 103
1xw3A 110 4240
1y6dA 114 5658
1y7xA 113 155
1ygmA 112 6
1ygtA 104 549
1ylmA 143 1498
1yn3A 98 15
1yrkA 126 1630
1ysyA 85 21
1yu5X 67 355
1yuaA 122 198
1yxeA 140 26
1z1zA 129 61
1z2fA 121 16
1z2uA 150 4155
1z4hA 66 8704
1z5fA 105 441
1z60A 59 254
1z8sA 146 2666
1zd0A 141 316
1zg2A 94 1950
1zo0A 126 140
1zt3A 80 741
1ztdA 125 13
1ztsA 139 105
1zw8A 64 32514
1zx8A 127 476
2a05A 57 125
2a2pA 129 115
2a7oA 100 52
2a7yA 80 110
2a8nA 130 7935
2a9iA 105 479
2aboA 131 3
2ahqA 67 911
2aivA 149 290
2aydA 76 1140
2ayyA 121 57
2b97A 70 99
2bicA 52 24
2bl5A 134 2243
2bw2A 140 123
2ccvA 99 229
2cg7A 90 51
2ch0A 133 191
2chhA 113 24
2ciuA 123 232
2cobA 70 6140
2cosA 54 30
2cprA 124 2698
2cqlA 100 2010
2cqyA 108 19011
2cr9A 139 537
2cwyA 94 465
2cx7A 129 1506
2cxcA 138 1611
2d0oB 108 148
2d46A 61 69
2d48A 129 49
2d56A 53 19
2d59A 141 26752
2d8sA 80 9401
2d96A 109 559
2d9kA 75 29742
2daeA 75 41
2di0A 71 241
2dipA 98 1566
2djfA 118 116
2djpA 77 6615
2dk4A 76 380
2dkyA 91 97
2dmwA 131 957
2dn8A 100 24517
2dogA 85 1726
2dp9A 124 570
2dqaA 123 134
2dsxA 52 1648
2dsyA 80 2573
2dt4A 143 938
2dzlA 66 911
2e0gA 107 1751
2e29A 92 147
2e60A 101 792
2e6jA 112 1888
2e6xA 69 14
2eeeA 149 2548
2efvA 82 18
2ehpA 124 3
2ejeA 114 100
2ejxA 134 21
2endA 137 67
2eo2A 71 295
2eqxA 105 2948
2es9A 100 16
2exnA 130 1209
2f3iA 150 232
2f60K 60 3988
2f6eA 125 3270
2fcjA 114 1377
2fhzA 106 13
2fhzB 93 66
2fj6A 82 205
2fj8A 120 145
2fkiA 118 1435
2fm4A 128 4218
2fmcA 82 24
2fqhA 109 11
2fygA 128 24
2fz0A 149 15
2g7jA 112 28
2gccA 63 1790
2gdtA 116 6
2ghfA 102 29832
2gjiA 82 6
2glwA 92 157
2gmoA 75 5
2gpiA 91 268
2gqbA 130 122
2gqcA 70 338
2griA 112 18
2h2mA 108 40
2h7aA 110 253
2h8eA 120 662
2hajA 135 351
2hbaA 52 1395
2hbpA 66 107
2hc5A 117 571
2hc8A 113 12799
2hf1B 59 1261
2hf6A 149 1528
2hg7A 60 21
2hgkA 117 242
2hh8A 127 95
2hi6A 132 1115
2hinA 66 13518
2hjjA 66 64
2hkvA 148 1281
2hl7A 82 645
2hngA 125 51
2hrvA 139 124
2hstA 136 796
2hwtA 94 585
2i5uA 77 764
2i9xA 86 264
2ia7A 111 1489
2iayA 114 95
2iblA 108 27
2igpA 114 212
2ikdA 56 465
2inwA 117 92
2iy2A 69 658
2iz3A 94 90
2j4mA 100 72
2j6aA 136 319
2j6bA 109 38
2j8jA 90 472
2jekA 140 266
2jhbA 143 59
2jmkA 110 3
2jmpA 100 1760
2jn6A 97 6087
2jneA 71 255
2jnsA 90 496
2joeA 139 145
2jonA 101 1005
2jorA 79 45
2joxA 106 34
2jozA 135 5
2jp0A 110 2785
2jpnA 79 19
2jr1A 79 878
2jr7A 84 424
2jrmA 60 111
2jroA 74 75
2js9A 81 819
2jsnA 96 41
2jtdA 122 47
2jtgA 87 1221
2jubA 76 2
2juoA 89 30
2jv2A 76 353
2jv8A 73 1
2jvfA 94 1
2jvmA 80 539
2jvuA 98 2
2jvwA 82 316
2jx3A 110 119
2jxtA 86 307
2jyeA 72 339
2jynA 146 166
2jysA 91 12
2jzfA 143 9
2k0mA 104 167
2k13X 103 1
2k32A 116 21944
2k3dA 87 369
2k3oA 129 43
2k47A 73 8
2k4eA 134 1624
2k4vA 125 47
2k4zA 125 2906
2k53A 70 931
2k5cA 88 828
2k5sA 73 83
2k7xA 120 32
2k87A 116 16
2k89A 80 235
2k8eA 116 297
2k8qA 134 1581
2k9aA 136 170
2k9hA 57 36
2kafA 67 3
2kbzA 99 326
2kcaA 109 763
2kcdA 120 15
2kd2A 94 81
2kd3A 97 28
2kdxA 119 816
2keoA 92 3832
2keyA 112 9095
2kfsA 146 693
2kfvA 73 29
2kgsA 132 2192
2kiaA 129 45
2kicA 96 36
2kieA 124 15
2kjqA 149 11673
2kk4A 95 4
2kkvA 121 8685
2kl5A 110 185
2kldA 73 15265
2km1A 117 103
2kmgA 142 190
2knqA 142 494
2ko6A 89 65
2konA 82 2
2kouA 102 311
2koyA 141 7256
2kptA 148 1778
2kq2A 147 4507
2kq9A 112 1586
2kqrA 113 279
2kreA 100 7704
2krtA 121 4
2krxA 94 85
2ksnA 128 161
2kswA 66 1
2kt7A 102 576
2kuoA 91 175
2kutA 122 608
2kvoA 120 139
2kvsA 80 201
2kvtA 71 25
2kwpA 129 1249
2kwqA 92 71
2kx7A 111 53
2kxsA 146 563
2kxyA 100 918
2kyuA 67 2878
2kz3A 83 203
2kz4A 112 774
2kzbA 114 12
2kzhA 134 1802
2kzxA 131 1360
2l0cA 97 169
2l0kA 93 25608
2l0rA 106 520
2l1iA 122 446
2l1nA 120 84
2l1tA 109 194
2l25A 141 148
2l32A 66 3397
2l3oA 124 15
2l42A 97 1405
2l48A 85 70
2l5fA 92 2383
2l69A 134 1
2l6oA 114 48
2l7kA 76 53
2l7pA 100 459
2l7sA 52 59
2l7yA 89 152
2l8kA 123 8
2l8oA 144 6104
2l9dA 108 124
2l9jA 119 7
2laiA 101 5
2lbfA 69 875
2lboA 123 26
2lcrA 97 652
2levA 57 860
2lezA 120 13
2lfeA 106 25
2lfpA 139 535
2lg7A 129 1
2lhnA 80 379

2lioA 136 2724
2lisA 131 34
2ljwA 104 103
2ljxA 82 260
2lklA 81 98
2ll0A 104 189
2llgA 143 96
2llvA 71 396
2llxA 142 248
2llzA 100 70
2ln7A 147 2162
2ln8A 75 15
2lnvA 104 540
2lpmA 123 58293
2lq6A 79 1009
2lqkA 70 1770
2lr4A 128 48
2lrdA 61 1
2lsjA 97 117
2lsmA 61 31
2lu2A 81 3
2lwdA 96 101
2lyxA 87 42
2mcmA 112 46
2nefA 136 1430
2nplX 96 22464
2nszA 129 446
2nwfA 141 7654
2o0qA 114 468
2o4tA 90 246
2o90A 115 1752
2oa4A 101 920
2od0A 103 777
2ofcA 141 41
2ofqA 95 599
2ohwA 128 176
2ookA 125 506
2ovsA 118 84
2oy9A 85 167
2p08A 107 302
2pjhA 76 1421
2pstX 61 531
2pvbA 107 13182
2pwwA 115 106
2pxgA 118 1302
2pyqA 114 110
2q03A 133 126
2qebA 145 818
2qkhA 94 620
2qq4A 138 1690
2qskA 95 6
2quoA 123 1
2ra9A 127 328
2rbgA 124 8
2rffA 111 3980
2rh3A 121 10
2rjiA 84 26
2rngA 79 16
2rqpA 88 921
2rqxA 81 48
2rrdA 101 2594
2rreA 74 47
2rrfA 141 36
2sakA 121 8
2sicI 107 188
2tgiA 112 986
2uwiA 127 1305
2v31A 112 408
2v33A 91 31
2v3sA 96 90
2v4xA 131 63
2v94A 96 361
2vh3A 112 1
2vkjA 106 14
2vlqA 84 174
2vn6B 64 806
2w0gA 123 190
2w0nA 118 18810
2w6kA 142 924
2wdsA 123 3733
2wtpA 93 242
2wzoA 146 444
2x43S 67 80
2x5cA 99 2
2x5rA 124 1
2x8nA 109 297
2xdgA 89 652
2xetA 89 1295
2xrhA 100 49
2xskA 95 32
2xv9A 134 53
2xw6A 130 2029
2xwsA 126 3080
2xz8A 135 1879
2yrcA 59 660
2yrgA 59 3441
2yruA 118 382
2yskA 144 349
2yukA 90 2124
2z5eA 122 155
2z5vA 141 3741
3a5pA 103 1
3ag3E 105 116
3agnA 114 168
3ajfA 92 6
3apaA 138 858
3ba3A 143 6192
3bqxA 139 20427
3bs1A 103 4167
3bv8A 85 187
3by8A 133 4081
3c0fB 85 14
3c5kA 108 1397
3c8iA 127 43
3ccdA 85 2277
3cjsB 72 862
3crdA 100 833
3ct6A 129 1965
3d7aA 137 241
3d7iA 97 5726
3db7A 127 538
3dsoA 66 14
3dt5A 119 1
3e11A 114 547
3e9vA 120 176
3eerA 140 3669
3eipA 84 32
3emiA 110 29
3eoiA 123 78
3f6qB 72 3761
3ft1A 100 885
3g1jA 90 19
3g2bA 90 1049
3ga8A 67 877
3he8A 148 1679
3hi2B 97 107
3hsaA 124 329
3idfA 138 13542
3k3vA 79 495
3k6iA 99 5037
3k7cA 108 144
3kbyA 145 22
3klrA 125 9705
3kwrA 83 2342
3lazA 96 768
3ld7A 87 383
3le4A 55 37
3lywA 86 792
3m1xA 126 6275
3maoA 105 2473
3mswA 139 74
3mx7A 90 69
3n01A 87 41
3njnA 114 22
3nklA 122 7246
3nohA 123 4
3nphB 131 508
3nrfA 102 20
3ny3A 70 328
3o0gD 149 43
3o2eA 86 1869
3obhA 66 153
3oblA 132 17
3obqA 141 3921
3od9A 122 43
3oizA 92 6907
3oj0A 138 39595
3ol3A 98 136
3ouvA 67 3555
3ovkA 129 3754
3owrA 127 273
3oxpA 148 4127
3p4hA 117 411
3pkzA 124 5457
3pmcA 132 21
3ql9A 125 2011
3qr7A 115 481
3qrlA 137 468
3qu3A 122 236
3r2cA 138 3174
3rmqA 111 26
3t3lA 121 514
3t7zA 119 10
3tekA 139 10
3tipA 132 1142
3tuoA 104 36
3tysA 75 35176
3u28C 92 306
3u3gA 140 5174
3u6gA 125 4
3u97A 77 839
3ultA 114 51
3upsA 108 1788
3v0rA 130 62
3vc8A 81 25
3vdjA 71 4012
3vk6A 96 3903
3zr8X 65 13
3zuaA 142 2897
3zyhA 121 23
3zzoA 93 48
4a94C 51 1
4b6iA 100 93
4dm5A 87 760
4dnyA 109 55
4e5xG 99 9
4e6kG 57 2634
4e6sA 85 742
4ebgA 97 42
4eqpA 129 2334
4ew7A 113 6
4exoA 146 4653
4f55A 128 923
4f98A 62 194
4g7xA 92 1
4g9sB 111 416
4goqA 108 177
4heiA 92 1792
4mt2A 61 113
4ubpA 100 571

C.2 Test set of EPC-map (EPC-map_test)

1a1xA 106 54	1bnoA 87 2047	1c7kA 132 1417	1couA 85 1714	1eaqA 124 100	1earA 142 551
1ezgA 82 75	1fadA 95 1174	1g03A 134 6	1gh8A 89 404	1ijyA 122 612	1kafA 108 15
1kn6A 73 250	1lmiA 131 72	1luzA 85 7786	1nz9A 58 3390	1oo0A 144 118	1pa4A 96 1641
1pcnA 93 179	1pucA 101 216	1qklA 127 273	1r26A 113 26039	1s7iA 124 2734	1se7A 83 188
1t0yA 90 4087	1u9dA 122 82	1ucrA 74 51	1uj8A 73 224	1v66A 65 65	1wi9A 72 113
1wijA 127 61	1wnaA 127 13	1wpiA 133 775	1xc5A 68 6036	1xn6A 143 7365	1yd0A 89 2606
1yr1A 119 686	1ywyA 74 25	1zmaA 118 23897	2ai6A 125 125	2ccqA 99 442	2cokA 113 3220
2cwrA 97 1086	2cxyA 114 800	2d58A 107 10221	2da4A 80 4860	2di7A 124 1001	2dirA 98 1494
2dstA 122 39035	2dy8A 69 2306	2e7mA 113 402	2eapA 90 333	2eqkA 85 1382	2f9hA 121 240
2fzpA 134 29	2gu3A 128 594	2h3jA 75 2565	2hqsC 107 8465	2jmlA 81 10005	2jn9A 105 14
2jobA 102 35	2jspA 87 506	2jw5A 106 2438	2jx0A 131 47	2k2bA 111 6	2k4nA 111 11
2k8oA 60 29	2kbiA 97 566	2kcnA 55 13	2khqA 110 12955	2kjpA 91 13650	2kkxA 102 42
2klqA 114 203	2knjA 90 36	2ko2A 67 121	2kp6A 79 197	2krkA 86 7254	2ktaA 74 561
2l3bA 130 96	2l3uA 98 867	2l5vA 150 204	2l76A 95 697	2lc0A 132 228	2lepA 63 498
2lfvA 106 1114	2lkgA 140 2023	2loeA 127 110	2lq7A 97 289	2lqlA 83 201	2ltlA 119 699
2lw6A 80 1	2lwxA 88 84	2lz0A 100 4237	2npbA 88 587	2nqwA 87 3881	2o37A 81 12261
2olmA 133 1706	2owaA 128 1689	2p9xA 98 8	2pspA 106 482	2qycA 102 1225	2rhkA 119 42
2v1nA 111 177	2vlqB 134 173	2wgoA 98 1	2xfvA 108 13	2xppA 137 188	2xuvA 69 103
2y78A 122 7390	2z4dA 96 203	2z59A 109 227	3a38A 83 179	3c9pA 122 224	3cptA 119 647
3ffyA 112 5297	3fymA 82 32215	3g21A 77 352	3glvA 122 10307	3goeA 80 733	3ldcA 82 5472
3lyyA 102 459	3mazA 99 3067	3ndqA 97 600	3nswA 106 1	3nzlA 73 28947	3ov5A 83 10
3r87A 132 7414	3rfeA 119 8707	3sxuB 102 152	3t7lA 74 2364	3ushA 118 326	4gioA 96 185