Consensus-based Control Over Wireless Channels

vorgelegt von
M. Sc.
Fabio Molinari
ORCID: 0000-0003-2617-962X

an der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
- Dr.-Ing. -

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr.-Ing. Marc Toussaint
Gutachter: Prof. Dr.-Ing. Jörg Raisch
Gutachter: Prof. Riccardo Scattolini
Gutachter: Prof. Dr.-Ing. Sławomir Stańczak

Tag der wissenschaftlichen Aussprache: 15. Dezember 2021

Berlin 2022

In memory of Mariuccia and Franco.

Abstract

Consensus is a useful concept in situations where a group of autonomous agents seek an agreement over a variable of common interest for the whole system. Agents communicate and, based on the information they collect from others, they take individual decisions according to the specific consensus protocol they implement. A consensus protocol that needs less resources for achieving a given result is more efficient. Efficiency in consensus has become a hot topic in research during the last years, mainly due to the rising number of applications that require battery-powered agents. This thesis proposes to increase the efficiency of consensus by exploiting the interference property of the wireless channel. This phenomenon has always been combatted. In fact, signals simultaneously transmitted by multiple agents in the same frequency are attenuated by unknown coefficients and summed up at the receiver. The standard approach is to discard such corrupted received signal, although it carries some useful information. In this thesis, we show that this corrupted signal can be actually used for achieving consensus. Under the assumption of a real-valued fading channel, we propose average consensus and max-consensus protocols that exploit interference. They both exhibit better performance than standard methods in terms of required wireless resources. We show that such protocols can be used for two practical problems, i.e., formation control of nonholonomic robots and distributively solving linear algebraic equations. This confirms the practical benefits of exploiting interference and motivates future experimental implementation.

Kurzfassung

Konsens spielt eine bedeutende Rolle in Szenarien in welchen mehrere autonome Agenten versuchen eine Einigung über eine für das ganze System wichtige Variable zu erzielen. Die zwei wesentlichen Phasen des Konsens sind Kommunikation und Berechnung, und zwar, aktualisiert jeder Agent seinen Standpunkt (durch ein sogenanntes "Konsensprotokoll"), auf Basis der von anderen Agenten erhaltenen Informationen. Ein Konsensprotokoll ist effizienter je weniger Ressourcen es braucht um ein bestimmtes Ziel zu erreichen. Effizienz des Konsensprotokolls hat in den letzten Jahren viel Aufmerksamkeit erlangt, hauptsächlich wegen des Wachstums der Akku-basierten Anwendungen. Im Rahmen dieser Doktorarbeit wird eine Methode präsentiert um die Effizienz des Konsensprotokolls durch die Ausnutzung der Funkinterferenz zu steigern. Interferenz in Funk-Netzwerken ist ein traditionell bekämpftes Phänomen. Die in der gleichen Zeit und Frequenz verbreiteten Signale werden erstens durch unbekannte Faktoren gedämpft und dann am Empfänger summiert. Traditionell bewertet man dieses geräuschvolle Signal als unbrauchbar, obwohl es einige Information trägt. Es wird demonstriert wie ein solches Signal benutzt werden kann um Konsens zu erreichen. Unter der Annahme eines reellwertigem fading Funknetz, wird ein Average-Konsens und ein Max-Konsens Protokol präsentiert, welches die Funkinterferenz ausnutzen. Die beiden Methoden zeigen bessere Leistung, im Hinblick auf nötige Ressourcen des Funknetz, im Vergleich zu Standardmethoden. In dieser Arbeit werden auch zwei mögliche praktische Implementationen von diesen Methoden gezeigt, und zwar Formationsregelung für die nicht-nonholonome Roboter und die verteilte Lösung von linearen algebraischen Gleichungen. Diese Ergebnisse bestätigen die Vorteile dieser Methode und motivieren künftige experimentelle Implementierung.

Acknowledgements

I am sincerely grateful to my advisor, Prof. Jörg Raisch, who believed in me and gave me, in a challenging moment of my career, the chance of joining his research group. His guidance and support have been significant and need to be acknowledged. My thanks for agreeing to chair the doctoral committee go to Prof. Sławomir Stańczak, who has helped me going through the communication-theoretical side of my research, and to Prof. Riccardo Scattolini, whose courses in *Politecnico di Milano* were the ones that shaped my desire and ambition to pursue a PhD. This thesis was possible thanks to the funding provided by the *Deutsche Forschungsgemeinschaft*, to which I am grateful.
My gratefulness is also towards all my colleagues, among whom I need to mention Dr. Thomas Seel, for his friendly and always genuine advice, Ms. Ulrike Locherer, for her assistance in bureaucratic and administrative topics, Navneet Agrawal, for having helped me understanding many topics in communication-theory, Dr. Christian Hans, for the kindness, help and availability, and Davide Zorzenon, for the collaboration in many projects. Some results presented in this thesis have also been achieved by cooperating with Dr. Alexander Katriniok, Prof. Alexandros Charalampidis, Zenit Music, Alexander Dethof, and Aaron Grapentin, to all of whom goes my gratitude.

There is a number of people who have contributed to my development, without whom this thesis would have been either different or, more probably, not possible.
My mother and father have raised me and given me qualities and defects. Through their austere but caring guidance, I developed that need of aiming always higher, of which I am grateful. A deep sense of gratitude is towards my brother Marco that has been a companion in the calm and in the rough seas, always rowing in my same direction, no matter if I were sitting next to him. I am grateful to my *nonni* that have taught me what they could, depending on their possibilities. My deep gratitude is for Klara for having handed me rope and gears during the climbing.
During my time in *Linz*, I met some people that have helped, supported, endorsed me, and to whom I am deeply grateful. Josef welcomed me in the city and proved to be a valuable friend. I need to mention the gratitude towards Danilo and Massimiliano for their important impact on my life. Many people in *Berlin* have contributed, each one under a different point of view, to open the doors of the city

to a foreigner like me. I am thankful to Alice and Kilian, for the possibility of seeing every morning the statue of *Klio* in *Nikolaivirtel*, and to Albrecht for having been the neighbor that everybody wants to have. My gratitude also to Amogh, a partner in many past, current and future business journeys. Among all *Berliners* that have spent with me the many days between my arrival and the completion of this thesis, I need to thank Michael, Alexander, Bianca, Nick, Philipp, Arnbjørn, Leonardo, Marco, Paul, Nicola, Thiago and Roberto. Every one with a different reason has contributed to make me feel at home. Although wandering across Europe, I have not forgotten my roots and the friends that, back in *Milano*, have given me more than I could ask. My long-time friends have stood behind me in good and also in bad decisions. I cannot be thankful enough to all of them, among whom I need to cite Matteo, Geppo, William, Andrea, Alessandro, and Umberto. I am also thankful, for the long-lasting friendship, to Mirko and Benedetta, schoolmates in those times when my days were filled with Latin and Philosophy, rather than differential equations. Finally, I am indebted to Simone and Alessio, deskmates during my time as an undergraduate, for having always stimulated me to do more, and with whom I have shared the decision of pursuing a PhD.

Finally, I need to underline my profound gratitude towards Germany and towards the *European Dream* for having allowed a son of Milan to become also a son of Berlin.

Contents

1
Introduction and Motivation

1.1 Motivation and Overview

Consensus is a useful notion in situations where several autonomous intercommunicating agents need to reach an agreement over a variable of common interest. An introduction to this topic can be found, e.g., in [92] and [21]. By decomposing the concept of consensus into its minimal terms, one obtains three core elements: a multi-agent system, a communication network, and an update protocol.
A **multi-agent system** is a collection of autonomous agents that need to collaborate (or compete) in order to achieve a common (or competitive) goal. For example, a swarm of quadcopters might be interested in achieving a formation around a target (see, e.g., [55]), a collection of autonomous ground vehicles could agree on who has to yield the way (see, e.g., [59]), or a group of traders could need to check the authenticity of a payment without the inclusion of any trusted intermediaries (see, e.g., [16]). All such depicted scenarios have in common the inherently distributed structure of the control action; the absence of a centralized all-knowing controller makes it fundamental that individual agents exchange information. In fact, each agent has access to a different point of view on the system. The goal/task cannot be achieved by an uncoordinated action on the environment. Coordination requires communication. For communication to take place, we need to establish a **communication network**, i.e., a structure that allows agents to communicate. Once agents are aware of others and of their points of view, they update their control action according to an **update protocol**. This is a qualitative description of how a multi-agent system can achieve consensus.

More formally, each agent has an estimate of a **variable of common interest** for the system (e.g., the list of vehicles ordered by their crossing priority), referred to as *information state*. By communicating, they get aware of information states of other agents and can finally update theirs. This procedure is iterated, until a consensus is eventually (possibly) achieved. The core of a consensus strategy is inside of the update protocol. Different protocols lead to different kinds of agreement. In fact, some might be interested in achieving a so-called average agreement, whilst someone else might be interested in the agreement on the maximum value (max-consensus). The fields of applica-

tion of such different protocols can be investigated trough examples: average consensus is a useful concept, e.g., for a platoon of vehicles on a lane that need to agree on a common lane speed (see [26]). On the other hand, max-consensus is a valid approach in a leader-election problem, e.g., if robots on a production line should decide which robot has to start the operation (see [17]).

In all such scenarios, energy consumption is one aspect that acquires high importance, especially in cases when agents are powered by batteries. In recent years, in fact, an increasing effort has been put towards the development of efficient consensus protocols, i.e., strategies that need less resources to let a system reach the agreement. For example, in [22] a data-quantization approach for communication is presented, in [4] the number of communication arcs is reduced, thus the communication effort, and in [57] the communication complexity is taken as a measure for the efficiency of a protocol. Similarly, the concept of optimal network topology is introduced in [85], thus arcs can be cut to increase protocol's efficiency; quantization is also used as a tool to reduce communication complexity, thus increasing efficiency, as in [8]. Approaches like these increase the efficiency of consensus protocols by changing some parameters at a communication level, but they all lie within the boundaries of **standard communication systems**, i.e., all methods that are standard practice in communication technology. In the context of wireless transmission, multiple users access the same wireless medium for transmitting over it, thus they need to share its capacity, see, e.g., [109, Ch. 7]. Communication systems that allow different users to share a common medium are referred to as *Multiple Access* techniques. Sharing a common medium is possible by assigning dedicated channels to multiple users by dividing the bandwidth. Two well-known approaches are based on frequency-division (FDMA) and time-division (TDMA). By [109, Ch. 7.4.1], in FDMA the system bandwidth is divided into orthogonal channels, nonoverlapping in frequency and allocated to the different users. In TDMA time is the "split resource" that is allocated to different users. Such multiplexing techniques allow independent transmissions of data, i.e., the signal (physically, the electromagnetic wave) carrying information coming from one source does not collide or interfere with other signals. Preventing interference (also called superposition) has some costs, but yields established results.

However, by looking at the interplay between *communication* and *control* at a lower level, [117] initiated a novel practice in the consensus community. In fact, it proposed a consensus protocol able to **exploit the interference property of the wireless channel**, rather than combatting it. This strategy does not rely on small changes of the communication system parameters (as the above mentioned papers), but proposes a completely new communication system. In fact, while traditional communication methods (e.g., TDMA, FDMA, etc) guarantee that every node-to-node transmission is free of interference, [103] shows that getting rid of interference is not a strict requirement for computing a function over the network. Instead of investing commu-

nication resources for combatting interference, a communication system can be designed so that interference is exploited, resources saved, and consensus reached.
The main topic of this thesis is the development of a family of consensus protocols able to handle interference. Unlike early attempts in the field (e.g., [35, 50, 107, 117]), we do not hold strict simplifying assumptions on the interference model (i.e., absence of fading and of additive noise). Moreover, we investigate both linear (i.e., average-) and nonlinear (i.e., max-) consensus protocols. Besides these theoretical findings, we propose the usage of such approaches for two practical problems: the solution of linear algebraic equations over the wireless channel and a consensus-based formation control approach for nonholonomic robots that aim at saving wireless resource by exploiting the superposition property of the channel.

This thesis is the sum of multiple works and scientific publications in the field (see Section 1.3), that span a time of three years, from the beginning of 2018 to the end of 2020. Such contributions have been published thanks to the support of the German Science Foundation (Deutsche Forschungsgemeinschaft), under their priority programme SPP-1914, "Cyber-physical networking".

1.2 Structure

Following this introduction, the thesis presents three main parts. Chapters 2-4 give some basic knowledge about consensus, its applications, and the communication methods. Average- and max-consensus over the wireless channels are the topic of Chapters 5-6. Finally, the usage of interference for reaching a formation in space is the topic of Chapter 7. A brief summary about each chapter is as follows:

- **Chapter 2**: The employed mathematical notation is explained. Prerequisites in linear algebra and matrix theory, together with properties of nonnegative matrices are reviewed. Standard approaches to average- and max-consensus are the topic of the last part of this chapter;
- **Chapter 3**: A context in which both average and max-consensus find practical application is the field of autonomous vehicles. First, we review how a consensus-based algorithm contributes to the automation of a road intersection. After that, both average and max-consensus are used for the automation of a highway;
- **Chapter 4**: We review standard wireless protocols and we briefly present the so-called Wireless Multiple Access Channel, which models the way multiple signals interfere at a receiver. This model will be the base for the development of our consensus protocols;
- **Chapter 5**: We present an average consensus protocol over fading wireless channels. Starting from such protocol, we present a strategy to solve linear algebraic equations by exploiting interference;

- **Chapter 6**: The topic of this chapter is a switching consensus protocol that exploits interference and allows the system to achieve max-consensus in finite-time;

- **Chapter 7**: We extend the average consensus protocol so that a group of robots moving in space can achieve a formation by exploiting the interference property of the channel. We propose a solution for agents having both single-integrator dynamics and non-holonomic dynamics;

- **Chapter 8**: This chapter contains final remarks and a final summary which also collects future work;

- **Appendix**: In the appendix we include some mathematical concepts that are of use throughout the thesis.

1.3 Related Publication of the Author

Some parts of this thesis have been extracted from the following publications, to all of which the author is the main contributor.

. **Molinari, Fabio**, Slawomir Stanczak, and Jörg Raisch. "Exploiting the superposition property of wireless communication for average consensus problems in multi-agent systems." In 2018 European Control Conference (ECC), pp. 1766-1772. IEEE, 2018.

. **Molinari, Fabio**, Sławomir Stańczak, and Jörg Raisch. "Exploiting the superposition property of wireless communication for max-consensus problems in multi-agent systems." IFAC-PapersOnLine 51, no. 23 (2018): 176-181.

. **Molinari, Fabio**, Navneet Agrawal, Slawomir Stanczak, and Jörg Raisch. "Max-Consensus Over Fading Wireless Channels." IEEE Transactions on Control of Network Systems (2021).

. **Molinari, Fabio**, and Jörg Raisch. "Exploiting wireless interference for distributively solving linear equations." IFAC-PapersOnLine 53, no. 2 (2020): 2999-3006.

. **Molinari, Fabio**, Alexander Katriniok, and Jörg Raisch. "Real-Time Distributed Automation Of Road Intersections." IFAC-PapersOnLine 53, no. 2 (2020): 2606-2613.

. **Molinari, Fabio**, and Jörg Raisch, "Efficient Consensus-based Formation Control With Discrete-Time Broadcast Updates," 2019 IEEE 58th Conference on Decision and Control (CDC), Nice, France, 2019, pp. 4172-4177.

. **Molinari, Fabio**, Aaron Grapentin, Alexandros Charalampidis, and Jörg Raisch. "Automating lane changes and collision avoidance on highways via distributed agreement." at-Automatisierungstechnik 67, no. 12 (2019): 1047-1057.

. **Molinari, Fabio**, and Jörg Raisch. "Automation of road intersections using consensus-based auction algorithms." In 2018 Annual American Control Conference (ACC), pp. 5994-6001. IEEE, 2018.

. **Molinari, Fabio**, Alexander Martin Dethof, and Jörg Raisch. "Traffic automation in urban road networks using consensus-based auction algorithms for road intersections." In 2019 18th European Control Conference (ECC), pp. 3008-3015. IEEE, 2019.

In the incipit of each chapter, we indicate the previous publication from which that chapter is extracted, if any. All such previous publications have the author as the main contributor.

2
Fundamentals

2.1 Preliminaries

Sets

Throughout this work, let $\mathbb{N}_0$, respectively $\mathbb{N}$, denote the set of nonnegative numbers, respectively positive integers. The set of real numbers, nonnegative real numbers, and positive real numbers are, respectively, $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{R}_{>0}$. The set of complex numbers is $\mathbb{C}$. Given a set of $m \in \mathbb{N}$ points in the n-dimensional space, i.e., $A = \{a_i\}_{i=1\ldots m} \subseteq \mathbb{R}^n$, $n \in \mathbb{N}$, its convex hull is $\mathfrak{C}(A)$. Formally,

$$\mathfrak{C}(A) = \{\sum_{i=1}^{m} \lambda_i a_i \mid \lambda_i \geq 0, \ \sum_{i=1}^{m} \lambda_i = 1\}.$$

The indicator function of a set $S \subseteq \mathbb{R}$ is denoted by $I_S : \mathbb{R} \mapsto \{0, 1\}$ and we have $\mathbf{I}_S(x) = 1$ if $x \in S$ and 0 otherwise.

Graphs

Let $\mathcal{G} := (\mathcal{N}, \mathcal{A})$ denote a graph, where $\mathcal{N} := \{1 \ldots n\}$ is the set of nodes, labeled 1 to $n \in \mathbb{N}$, and $\mathcal{A}$ is the set of arcs; the definition of this latter depends on whether $\mathcal{G}$ is *directed* or *undirected*. Let $\mathcal{A}_u$ be

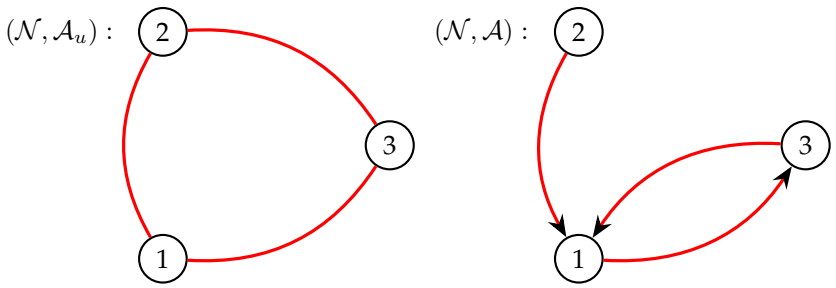


Figure 2.1: Undirected and directed graphs.

the set of arcs of an undirected graph. We have $\mathcal{A}_u \subseteq [\mathcal{N}]^2$, where $[\mathcal{N}]^2$ represents the set of all two-elements subsets of $\mathcal{N}$. Accordingly, $\{i, j\} \in \mathcal{A}_u$ if an (undirected) arc exists between nodes $i, j \in \mathcal{N}$.
For a directed graph, the arc set is $\mathcal{A} \subseteq \mathcal{N} \times \mathcal{N}$, namely $(i, j) \in \mathcal{A}$ if

a (directed) arc goes from node $i \in \mathcal{N}$ to $j \in \mathcal{N}$. In a directed graph, for $i \in \mathcal{N}$, (i, i) is called self-arc of node i. In a directed graph, if there exists an arc (i, j), i is a *parent* node of node j, and j is a *child* node of node i. The set of neighbor nodes of node $i \in \mathcal{N}$, denoted $\mathcal{N}_i \subseteq \mathcal{N}$, is the set of all nodes connecting to i. In a directed graph, $\mathcal{N}_i$ is the set of all parents of node i. Formally, the neighbor set is defined for undirected and directed graphs, as, respectively,

$$\mathcal{N}_i := \{j \in \mathcal{N} \mid \{i, j\} \in \mathcal{A}_u\} \tag{2.1}$$

and

$$\mathcal{N}_i := \{j \in \mathcal{N} \mid (j, i) \in \mathcal{A}\}. \tag{2.2}$$

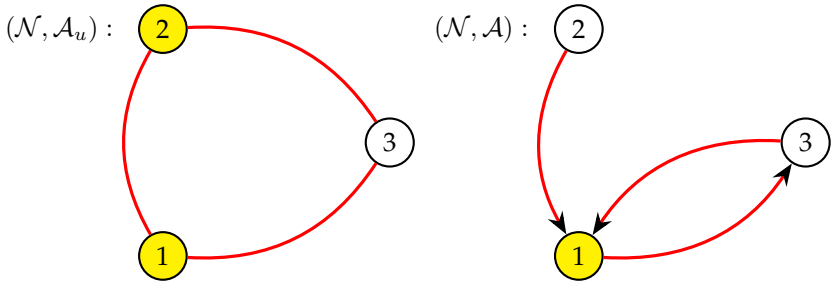


Figure 2.2: Neighbors of node $i = 3$ are highlighted in yellow.

In many contexts, time-varying graphs are considered. Let

$$\mathfrak{G} := \{\mathcal{G}(k)\}_{k \in \mathbb{N}_0} = \{(\mathcal{N}, \mathcal{A}(k))\}_{k \in \mathbb{N}_0}$$

be a sequence of graphs where $k \in \mathbb{N}_0$ is the iteration index and $\mathcal{G}(k)$ is the graph at iteration $k \in \mathbb{N}_0$. In the sequence $\mathfrak{G}$, the set of arcs varies at every iteration, i.e., $(i, j) \in \mathcal{A}(k)$ if a directed arc goes from node $i \in \mathcal{N}$ to node $j \in \mathcal{N}$ at iteration $k \in \mathbb{N}_0$ (the undirected case can be easily obtained). The respective set of neighbors of agent $i \in \mathcal{N}$ at iteration $k \in \mathbb{N}_0$ is $\mathcal{N}_i(k)$. A sequence of time-varying weighted directed graphs is

$$\mathfrak{G}_w := \{\mathcal{G}_w(k)\}_{k \in \mathbb{N}_0} = \{(\mathcal{N}, \mathcal{A}(k), \mathcal{W}(k))\}_{k \in \mathbb{N}_0},$$

where the set of arcs' weights $\mathcal{W}(k)$ is defined as

$$\mathcal{W}(k) := \{w_{i,j}(k) \in \mathbb{R}_{>0} \mid (i, j) \in \mathcal{A}(k)\}.$$

The undirected case can be easily obtained.
Let a path be defined as a sequence of nodes, such that each adjacent pair is connected by an arc. An undirected (directed) graph is connected (strongly connected) if a path exists between each pair of nodes. A directed, respectively undirected, graph $\mathcal{G}(k)$ is fully connected if,

$$\forall i \in \mathcal{N},\ \forall j \in \mathcal{N} \setminus \{i\},\ (i, j) \in \mathcal{A}(k),$$

respectively $\{i, j\} \in \mathcal{A}_u(k)$. A *rooted directed tree* is a directed graph in which every node has exactly one parent apart from one node, i.e.,

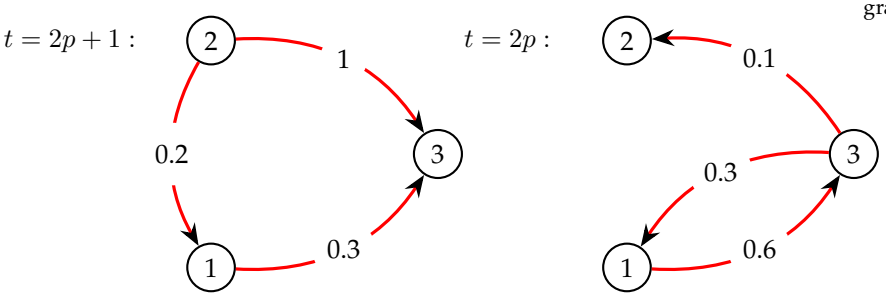


Figure 2.3: Sequence of weighted digraphs for $p = 0, 1, \ldots$.

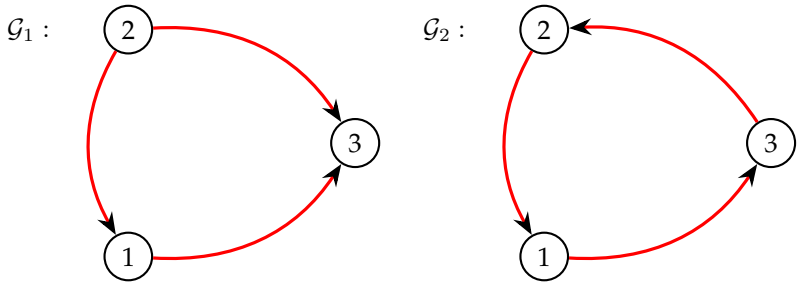


Figure 2.4: Strongly connected ($\mathcal{G}_2$) and not strongly connected ($\mathcal{G}_1$) graphs.

the *root node*, that has no parents but has a directed path to every other node. A directed graph $\mathcal{G}$ contains a *directed rooted spanning tree* if and only if $\mathcal{G}$ has at least one node with a directed path to all other nodes. A directed graph is balanced if the sum of incoming arcs' weights and outgoing arcs' weights corresponding to each node are equal, i.e., if, $\forall i \in \mathcal{N}$,

$$\sum_{j=1}^{n} w_{i,j} = \sum_{j=1}^{n} w_{j,i}.$$

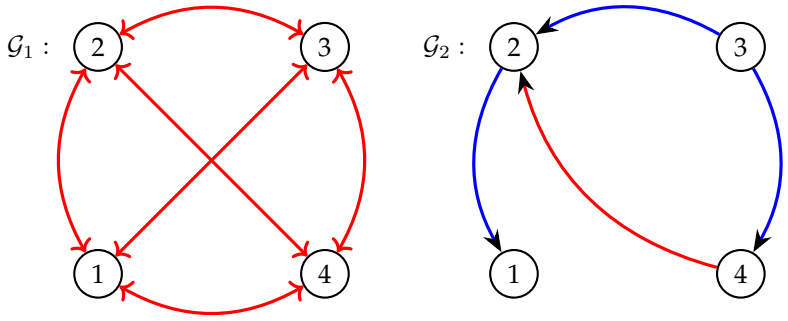


Figure 2.5: A fully connected directed graph ($\mathcal{G}_1$) and a directed graph containing a directed rooted spanning tree highlighted in blue ($\mathcal{G}_2$).

Two directed graphs with the same vertex set $\mathcal{N}$, i.e., $\mathcal{G}(1)$ and $\mathcal{G}(2)$, are given. Their composition, namely $\mathcal{G}(2) \circ \mathcal{G}(1)$, is a graph with the same vertex set $\mathcal{N}$ and an arc set defined such that (i, j) is an arc if and only if $(i, \ell) \in \mathcal{G}(1)$ and $(\ell, j) \in \mathcal{G}(2)$. Let $\mathfrak{G}$ be a finite sequence of $q \in \mathbb{N}$ directed graphs. The sequence is jointly strongly connected if $\mathcal{G}(q) \circ \mathcal{G}(q-1) \circ \cdots \circ \mathcal{G}(1)$ strongly connected. Let now $\mathfrak{G}$ be an infinite sequence of graphs with the same vertex set. The sequence is repeatedly jointly strongly connected if there exists $p \in \mathbb{N}$ such that

each finite sequence $\mathcal{G}(p(k+1)) \circ \cdots \circ \mathcal{G}(pk+1)$ is strongly connected.

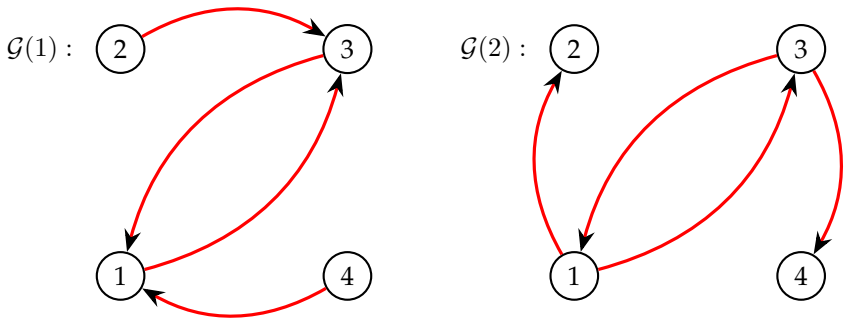


Figure 2.6: Jointly strongly connected sequence of graphs. $\mathcal{G}(1)$ and $\mathcal{G}(2)$ are individually not strongly connected, but their sequence is jointly strongly connected.

Matrices

Let $A \in \mathbb{R}^{n\times m}$ be a matrix of real numbers, where $n, m \in \mathbb{N}$. The entry in position (i, j), $1 \leq i \leq n$, $1 \leq j \leq m$, is $[A]_{ij}$. The transpose of A is denoted A'. The spectral radius of a square matrix A is denoted $\rho(A)$. The n-dimensional column vector of ones is $\mathbf{1}_n$ and the column vector of zeros is $\mathbf{0}_n$. The n-dimensional identity matrix is $\mathbb{I}_n$. Given a vector $\boldsymbol{v}_n$, its entry in position i is $[\boldsymbol{v}_n]_i$, $1 \leq i \leq n$. The $n \times n$ matrix whose diagonal entries are the entries of $\boldsymbol{v}_n$ (and off-diagonal entries are zero) is diag $(\boldsymbol{v}_n)$.

Definition 1. *Matrix $A \in \mathbb{R}^{n\times m}$ is nonnegative (positve) if each entry is nonnegative (positive).*

Definition 2. *The nonnegative matrix $A \in \mathbb{R}^{n\times n}$ is row-stochastic if each row sums up to* 1.

Definition 3. *The nonnegative matrix $A \in \mathbb{R}^{n\times n}$ is column-stochastic if each column sums up to* 1.

A matrix both row- and column-stochastic is called doubly-stochastic. Given two nonnegative matrices of the same dimension, namely A and B, we say that A is of the same type of B, and we write $A \sim B$, if nonzero entries of A and B are in the same positions. Given matrix $A \in \mathbb{R}^{n\times m}$, its kernel (or null space) is

$$\ker(A) := \{v \in \mathbb{R}^m \mid Av = \mathbf{0}_n\},$$

its nullity is denoted by nullity(A) and is defined as the dimension of $\ker(A)$, and its rank is denoted by rank(A) and is the dimension of the vector space spanned by its columns.

2.2 Properties of Nonnegative and Positive Matrices

Definition 4 (Adjacency Matrix). *The* adjacency matrix $A \in \mathbb{R}^{n\times n}$ *of a directed weighted graph* $\mathcal{G}_w$ *with node set* $\mathcal{N} = \{1, \dots, n\}$ *is defined through its elements such that,* $\forall i, j \in \mathcal{N}$, $[A]_{ij} = w_{j,is}$ *if* $(j, i) \in \mathcal{A}$, *otherwise* $[A]_{ij} = 0$.

Definition 5 (Laplacian). *Let* $A \in \mathbb{R}^{n\times n}$ *be the adjacency matrix of a directed weighted graph. Then, the Laplacian of the graph,* $\mathcal{L} \in \mathbb{R}^{n\times n}$, *is defined by*

$$\mathcal{L} := \mathcal{D} - A,$$

where $\mathcal{D}$ *is the degree matrix, which is a diagonal matrix whose diagonal elements are,* $\forall i \in \mathcal{N}$,

$$[\mathcal{D}]_{ii} := \sum_{j\in\mathcal{N}} [A]_{ij}.$$

Definition 6 ([37, Definition 6.2.11]). *A nonnegative matrix* $A \in \mathbb{R}^{n\times n}$ *is given. Then* $\Gamma(A)$ *is defined to be the weighted directed graph with adjacency matrix* A.

Definition 7 ([37, Definition 6.2.21]). *A matrix* $A \in \mathbb{R}^{n\times n}$ *is* reducible *if there exists a permutation matrix* Π *such that* $\Pi \cdot A \cdot \Pi'$ *is in block upper triangular form.*
A matrix is irreducible *if it is not reducible.*

Proposition 1 ([37, Theorem 6.2.14]). *The graph* $\Gamma(A)$ *with* A *nonnegative is strongly connected if and only if* A *is irreducible.*

Definition 8 ([37, Theorem 8.5.2]). *The nonnegative matrix* $A \in \mathbb{R}^{n\times n}$ *is* primitive *if and only if* A^m *is positive for some* $m \geq 1$.

The exponent to which a nonnegative matrix can be elevated so as to obtain a positive matrix has been determined by Wielandt.

Theorem 1 ([37, Corollary 8.5.8]). *Let* $A \in \mathbb{R}^{n\times n}$ *be nonnegative. Then,* A *is primitive if and only if* A^{n^2-2n+2} *is positive.*

Corollary 1 ([37, Lemma 8.5.4]). *An irreducible matrix with positive diagonal is primitive.*

The following well-known result, the so-called *Perron-Frobenius Theorem*, helps determining the convergence conditions of primitive matrices.

Theorem 2 ([37, Theorem 8.2.8, Theorem 8.5.1, pg. 516]). *Let* $A \in \mathbb{R}^{n\times n}_{>0}$ *be a primitive matrix. Then*

1. $\rho(A) > 0$
2. $\rho(A)$ *is an algebraic simple eigenvalue of* A
3. *There is a unique real positive vector* $\boldsymbol{v}_n$ *such that* $A\boldsymbol{v}_n = \rho(A)\boldsymbol{v}_n$
4. *There is a unique real positive vector* $\boldsymbol{w}_n$ *such that* $\boldsymbol{w}_n' A = \rho(A)\boldsymbol{w}_n'$ *and* $\boldsymbol{w}_n'\boldsymbol{v}_n = 1$

5. $|\lambda| < \rho(A)$ *for every eigenvalue* λ *of* A *such that* $\lambda \neq \rho(A)$
6. $(\rho(A)^{-1}A)^m \to \boldsymbol{v}_n \boldsymbol{w}_n'$ *as* $m \to \infty$.

Another useful tool for analyzing the spectrum of products of primitive matrices is the so-called *Gersgorin Theorem*.

Theorem 3 ([31, Theorem 6.8])**.** *Every eigenvalue* λ *of a real* $n \times n$ *matrix* A *satisfies at least one of the inequalities*

$$|\lambda - [A]_{ii}| \leq \sum_{\substack{j=1 \\ j \neq i}}^{n} |[A]_{ij}|,$$

$i = 1, \ldots, n$.
The circles on the complex plane centered in $[A]_{ii}$ *with radius*

$$\sum_{\substack{j=1 \\ j \neq i}}^{n} |[A]_{ij}|,$$

$\forall i = 1 \ldots n$, *are the* Gershgorin discs *of matrix* A. *One can see that all eigenvalues of matrix* A *lie inside of the union of its Gershgorin discs.*

Proposition 2. *Given two row-stochastic matrices, their product is also row-stochastic.*

Proof. Let A and B be row-stochastic. Their product is defined through its elements as, $\forall i, j \in [1, n]$,

$$[AB]_{ij} = \sum_{k=1}^{n} [A]_{ik}[B]_{kj}.$$

The row-sum of AB is, $\forall i \in [1, n]$,

$$\sum_{j=1}^{n} [AB]_{ij} = \sum_{j=1}^{n} \sum_{k=1}^{n} [A]_{ik}[B]_{kj} = \sum_{k=1}^{n} [A]_{ik} (\sum_{j=1}^{n} [B]_{kj}) = \sum_{k=1}^{n} [A]_{ik} = 1.$$

This concludes the proof. □

The convergence properties of products of primitive matrices can also be characterized thanks to the contributions of Wolfowitz, see [113]. Given a row-stochastic matrix P, let $\delta(P)$ be a measure of how different the rows of P are (see [113]). Let $A_1, \ldots, A_k$ be row-stochastic matrices of the same dimension. Any product $A_{i_1} \cdot \cdots \cdot A_{i_\ell}$, $\{i_1, \ldots, i_\ell\} \subseteq \{1, \ldots, k\}$, $\ell \geq 1$, is called *word in the* A*'s*.

Theorem 4 ([113, Theorem 1])**.** *Let* $A_1, \ldots, A_k$ *be row-stochastic matrices of the same dimension, such that any product combination of them is primitive*[1] *and row-stochastic. For any* $\epsilon > 0$, *there exists* $\nu(\epsilon) \in \mathbb{N}$ *such that* $A_{i_1} \cdot \cdots \cdot A_{i_{\nu(\epsilon)}}$, $\{i_1, \ldots, i_{\nu(\epsilon)}\} \subseteq \{1, \ldots, k\}$, *satisfies*

$$\delta(A_{i_1} \cdot \cdots \cdot A_{i_{\nu(\epsilon)}}) < \epsilon.$$

This theorem shows that any sufficiently long product in the A's has all its rows approximately the same. To this end, it is sufficient to

[1] Wolfowitz's results apply for *indecomposable, aperiodic, stochastic* matrices (SIA). [97, pg. 147] claims that Markov proved that SIA matrices and primitive matrices have the same properties.

require any product in the A's to be primitive. Theorem 4 implies that an infinite product of primitive row-stochastic matrices converges to a matrix whose rows are all identical, i.e.

$$\lim_{k\to\infty} A_k A_{k-1} \cdot \cdots \cdot A_1 A_0 = \mathbf{1}_n \boldsymbol{r}_n', \tag{2.3}$$

where $\boldsymbol{r}_n \in \mathbb{R}^n$. It is also possible to characterize more precisely this vector $\boldsymbol{r}_n$. In fact, by Proposition 2, $\mathbf{1}_n \boldsymbol{r}_n'$ is also row-stochastic, since product of row-stochastic matrices. Thus, $\boldsymbol{r}_n$ is such that $\boldsymbol{r}_n' \mathbf{1}_n = 1$, namely, $\sum_{i=1}^n [\boldsymbol{r}_n]_i = 1$.

2.3 Standard Approaches to Consensus

A multi-agent system is a collection of distinct agents that need to collaborate to achieve some global tasks. Each agent has the individual capability of dealing with the surrounding environment, be it moving in space or performing some specific actions. Inter-agent cooperation is enabled by letting individual agents exchange information with each other. Thus, the underlying network topology, i.e., the analytical structure of the communication system, can be modeled by a graph (be it invariant or variant through time). In particular, for an agent set $\mathcal{N} = \{1, \dots, n\}$, where agents are labeled 1 to n, the directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ models the underlying network topology, i.e., each directed arc (i, j) in $\mathcal{A}$ is defined (exists) if agent $i \in \mathcal{N}$ transmits information to agent $j \in \mathcal{N}$. Information is exchanged at discrete-time steps, although in literature also continuous-time exchange of information is presented.

To perform a cooperative task, agents are required to share a common view of both the environment and the goal. Thus, achieving an agreement on entities of common interest (normally variables) is paramount for any multi-agent system. Consensus is a useful tool towards this end. In fact, consensus guarantees that individual agents sharing information with each other get to an agreement that is critical for achieving the common goal. Thorough summaries of the topic are in [92], [89], and [90]. Each agent has a local estimation of the variable of common interest that needs to be agreed upon. This local guess is called *information state* and it is commonly denoted by x_i (for agent $i \in \mathcal{N}$). The information state dynamics is the way the local information state is updated according to the information received from other agents. This is precisely the *consensus protocol*.

In what follows, we review two main families of consensus protocols: average- and max-consensus. As it will be carefully explained in Section 4, traditional communication protocols are employed, so that, if $(i, j) \in \mathcal{A}$, then j can retrieve the information of agent i. *Orthogonal channel access methods* allow a receiver to individually reconstruct the information sent by each transmitter. The topic of communication technology will be the focus of Section 4.

2.3.1 Average consensus

Continuous time

If the communication network allows continuous communication or if the bandwidth is sufficiently large, then the formalism of a continuous-time consensus dynamics can be employed. The underlying network topology is represented (in the general case) by a time-varying weighted directed graph, i.e., $\forall t \in \mathbb{R}_{\geq 0},\ (\mathcal{N}, \mathcal{A}(t), \mathcal{W}(t))$. The most common continuous time average-consensus protocol in literature (see [29, 78,

92]) is, $\forall i \in \mathcal{N}$,

$$\forall t \in \mathbb{R}_{\geq 0},\ \dot{x}_i(t) = -\sum_{j=1}^{n} a_{ij}(t)(x_i(t) - x_j(t)), \tag{2.4}$$

where $a_{ij}(t)$ is the (i,j) entry of the adjacency matrix of graph $(\mathcal{N}, \mathcal{A}(t), \mathcal{W}(t))$. By definition of neighbors set and adjacency matrix, $a_{ij}(t) \neq 0$ if and only if $j \in \mathcal{N}_i(t)$. By this, (2.4) can be rewritten as

$$\forall t \in \mathbb{R}_{\geq 0},\ \dot{x}_i(t) = -\sum_{j \in \mathcal{N}_i(t)} a_{ij}(t)(x_i(t) - x_j(t)). \tag{2.5}$$

By this consensus protocol, it turns out that the information state $x_i(t)$ is driven towards the information states of i's neighbors. This leads to the conclusion that $x_i(t)$ remains constant (i.e., we have equilibrium) at time $t \in \mathbb{R}_{\geq 0}$ if

$$\forall j \in \mathcal{N}_i(t),\ x_i(t) = x_j(t), \tag{2.6}$$

i.e., when i and all its neighbors share the same local estimation of the variable of common interest[2]. If this is the case for all agents, consensus is achieved. Formally, this is true if

[2] That is to say that they have the same information state.

$$\exists t \in \mathbb{R}_{\geq 0} :\ \forall i, j \in \mathcal{N},\ x_i(t) = x_j(t) = x^*. \tag{2.7}$$

The conditions for which (2.7) eventually applies have been investigated in literature (see [92] for a complete summary) and will be here reviewed. To this end, a common tool employed for convergence analysis comes directly from matrix theory. Let, $\forall t \in \mathbb{R}_{\geq 0}$, $\boldsymbol{x}(t) \in \mathbb{R}^n$ stack all information states, i.e.,

$$\forall i \in \mathcal{N},\ [\boldsymbol{x}(t)]_i = x_i(t). \tag{2.8}$$

Dynamics (2.5) can be rewritten in the corresponding matrix form as

$$\dot{\boldsymbol{x}}(t) = -\mathcal{L}(t)\boldsymbol{x}(t), \tag{2.9}$$

where, $\forall t \in \mathbb{R}_{\geq 0}$, $\mathcal{L}(t)$ is the *Laplacian* of graph $(\mathcal{N}, \mathcal{A}(t), \mathcal{W}(t))$.

In case the arcs set is constant, $\mathcal{L}(t)$ is time-invariant, i.e., $\forall t \in \mathbb{R}_{\geq 0}$, $\mathcal{L}(t) = \mathcal{L}$; system (2.9) becomes, $\forall t \in \mathbb{R}_{\geq 0}$,

$$\dot{\boldsymbol{x}}(t) = -\mathcal{L}\boldsymbol{x}(t), \tag{2.10}$$

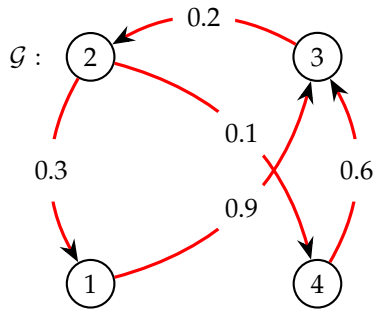
so that

$$\lim_{t \to \infty} \boldsymbol{x}(t) = \lim_{t \to \infty} e^{-\mathcal{L}t}\boldsymbol{x}(0).$$

When talking about eigenvalues, for (2.7) to hold for $t \to \infty$, $\mathcal{L}(t)$ needs to have one eigenvalue in the origin, with all other eigenvalues in the left half-plane. The following proposition then provides a sufficient condition for this.

Proposition 3 ([79, Theorem 1.i])**.** *If the underlying network topology is a strongly connected weighted digraph, then the system achieves consensus, i.e., (2.7) holds for $t \to \infty$.*

Definition 9. *The* linear average *of a set of values is the arithmetic mean of all the values. The* weighted average *is the convex combination of all the values, where all coefficients are non-negative and sum to 1.*



$$A = \begin{bmatrix} 0 & 0.3 & 0 & 0 \\ 0 & 0 & 0.2 & 0 \\ 0.9 & 0 & 0 & 0.6 \\ 0 & 0.1 & 0 & 0 \end{bmatrix}$$

Figure 2.7: Underlying network topology and corresponding adjacency matrix for Example 1.

Example 1. *A multi-agent system composed of n agents is given where $n = 4$. The underlying network topology is time-invariant and is depicted in Figure 2.7. The initial information states vector is*

$$\boldsymbol{x}(0) = [0.5, 2, -1, 0.1]'.$$

All agents run (2.5). Consensus is asymptotically achieved at $x^* = 0.92$ and the result is illustrated in Figure 2.8. Note that the agreement value is not the linear average of the initial values.

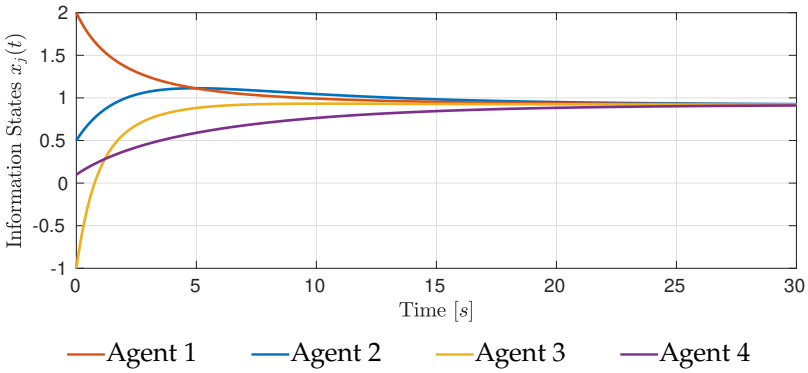


Figure 2.8: Asymptotic convergence to the consensus for Example 1.

However, having a strongly connected topology is not the strictest condition. In fact, it has been proven in [91] that the necessary and sufficient graph condition for (2.9) to achieve consensus is that the underlying network topology should contain a *rooted directed spanning tree*[3]. This is formalized by the following proposition.

[3] Whose meaning has been clarified in the previous section.

Proposition 4 ([91, Theorem 2])**.** *System (2.9) achieves consensus if and only if the underlying network topology contains a directed rooted spanning tree.*

Example 2. *The topology in Figure 2.9 contains a directed rooted spanning tree, with node 2 as root. Initial conditions are as in Example 1. Node 2 has no parents, thus it does not receive any information from other nodes. This implies that all agents will eventually reach the same state equal to $x_2(0)$. The contributions of all other initial information states will be lost. This is illustrated in Figure 2.10.*

Remark 1. *Although the graph being a directed rooted spanning tree constitutes the formal necessary and sufficient condition for consensus, we consider*

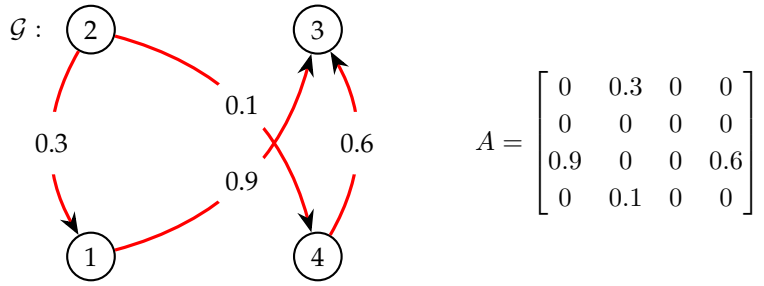


Figure 2.9: Underlying network topology and corresponding adjacency matrix for Example 2.

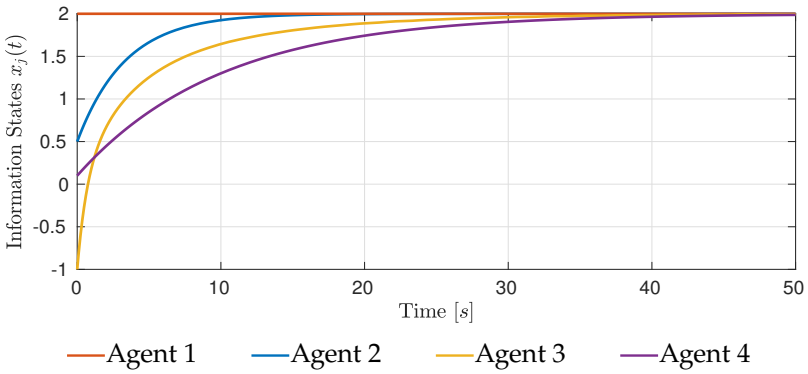


Figure 2.10: Asymptotic convergence to the consensus for Example 2.

this condition irrelevant for the problem at hand. In fact, in the case the graph itself is a directed rooted spanning tree, the root-agent will eventually spread its initial information state across the network. This is a degeneration of the concept of distributed agreement, since it does not involve any negotiation with other agents, but only propagation of information. A

Because of the latter remark, throughout this paper, we will not consider the case of a directed rooted spanning tree, but we will rather focus on strong connectedness as a sufficient condition for consensus.

Proposition 5 ([79, Theorem 1.ii]). *The consensus value of system (2.10) is*

$$x^* = \boldsymbol{w}_n \boldsymbol{x}(0),$$

where $\boldsymbol{w}_n$ is the left eigenvector of $\mathcal{L}$ associated with the eigenvalue $\lambda = 0$.[4]

Definition 10. *A weighted directed graph is* balanced *if, for each node, the sum of the weights of the arcs outgoing from that node is equal to the sum of the weights of the arcs incoming to that node.*

Proposition 6 ([79, Theorem 1.iii]). *If the underlying network topology is a balanced digraph, the consensus value of system (2.10) is the linear average of initial information states, i.e.,*

$$x^* = \frac{1}{n} \mathbf{1}_n' \boldsymbol{x}(0).$$

In case the arcs set changes with time, the underlying topology $\mathcal{L}(t)$ is time-variant; in most literature this topic is addressed as *consensus in switching networks*, see [79, Sec.III]. [78, 79] contain convergence results for this case. It is, in fact, typical to consider the communication topology to be *piecewise constant* over finite lengths of time (referred to

[4] By [79, Theorem 1], if the underlying network topology is strongly connected, its corresponding Laplacian has a unique real eigenvalue in $\lambda = 0$.

as *dwell times*). These are lower bounded by a positive constant. This yields that $\mathcal{L}(t)$ is piecewise constant with dwell times $\tau_k = t_{k+1} - t_k$, $k \in \mathbb{N}_0$. By (2.9),

$$\lim_{t\to\infty} \boldsymbol{x}(t) = \lim_{k\to\infty} e^{-\mathcal{L}(t_k)\tau_k} e^{-\mathcal{L}(t_k-1)\tau_{k-1}} \cdot \cdots \cdot e^{-\mathcal{L}(t_0)\tau_0} \boldsymbol{x}(t_0). \quad (2.11)$$

The following assumption is typically formulated in literature.

Assumption 1 ([89, Sec. 2.3.1]). *Dwell times are nonzero and uniformly lower bounded.*

Remark 2 ([92]). *If the topology at dwell time t_k is strongly connected, $e^{-\mathcal{L}(t_k)\tau_k}$ is a row-stochastic, nonnegative, and irreducible matrix with positive diagonal.*

The latter remark shows that, by (2.11) and Corollary 1, the convergence analysis of *consensus in switching networks* reduces to the study of products of primitive row-stochastic matrices.

Theorem 5. *If the underlying network topology at every dwell time t_k, $k \in \mathbb{N}_0$, is strongly connected, then system (2.9) achieves consensus, i.e., (2.7) holds for $t \to \infty$.*

Proof. By Remark 2 and by the hypotheses of the theorem, $\forall k \in \mathbb{N}_0$, $\mathcal{L}(t_k)$ is row-stochastic, nonnegative, and primitive. By Theorem 4 and (2.3),

$$\lim_{k\to\infty} e^{-\mathcal{L}(t_k)\tau_k} e^{-\mathcal{L}(t_k-1)\tau_{k-1}} \cdot \cdots \cdot e^{-\mathcal{L}(t_0)\tau_0}$$

converges to a square matrix of rank 1 (all rows are identical), i.e., (2.7) holds. This formalizes the case in which we have an infinite number of switchings. On the other hand, in case we have a finite number of switchings, the last interval is infinite, and this trivially reduces to the time-invariant case. This concludes the proof. □

Example 3. *A sequence of randomly generated strongly connected weighted digraphs, say $\mathfrak{G}$, is employed. Switchings are shown in Figure 2.11. The network is composed of 4 agents with the same initial information states as Example 1. Results are illustrated in Figure 2.11.*

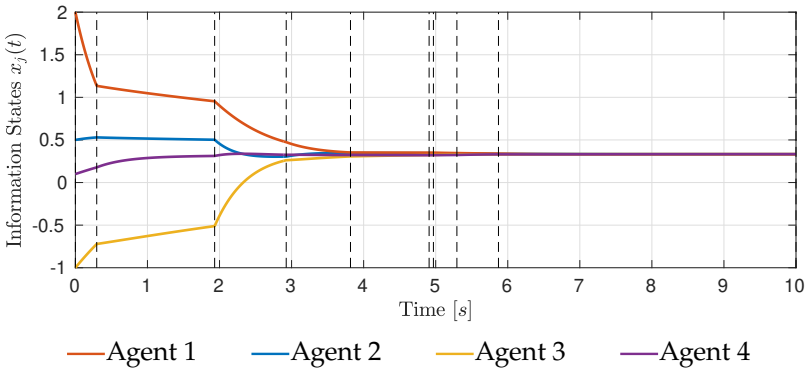


Figure 2.11: Asymptotic convergence to the consensus for Example 3. Switch times are highlighted by dashed lines.

However, a sequence of strongly connected topologies is not the strictest condition for consensus.

Theorem 6. *System (2.9) achieves consensus if the sequence of underlying network topologies is repeatedly jointly strongly connected, i.e., (2.7) holds.*

Proof. The proof follows directly from [89, Theorem 2.3.1]. □

Example 4. *The employed sequence of weighted digraphs, say $\mathfrak{G}$, is randomly generated to be p-repeatedly strongly jointly connected, with $p = 4$. The network is composed of 4 agents with the same initial information states as Example 1. Results are illustrated in Figure 2.12. Agent 3 (yellow) does not have any parent node in the communication graph until the third dwell time (dashed line); in correspondence to this, in fact, agent 3's information state is driven towards another agent's. The underlying topologies during the first 4 dwell times are illustrated in Figure 2.13.*

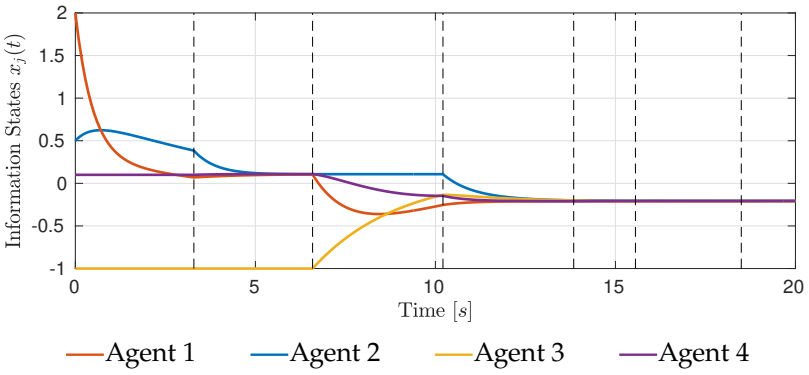


Figure 2.12: Asymptotic convergence to the consensus for Example 4. Dwell times are highlighted by dashed lines.

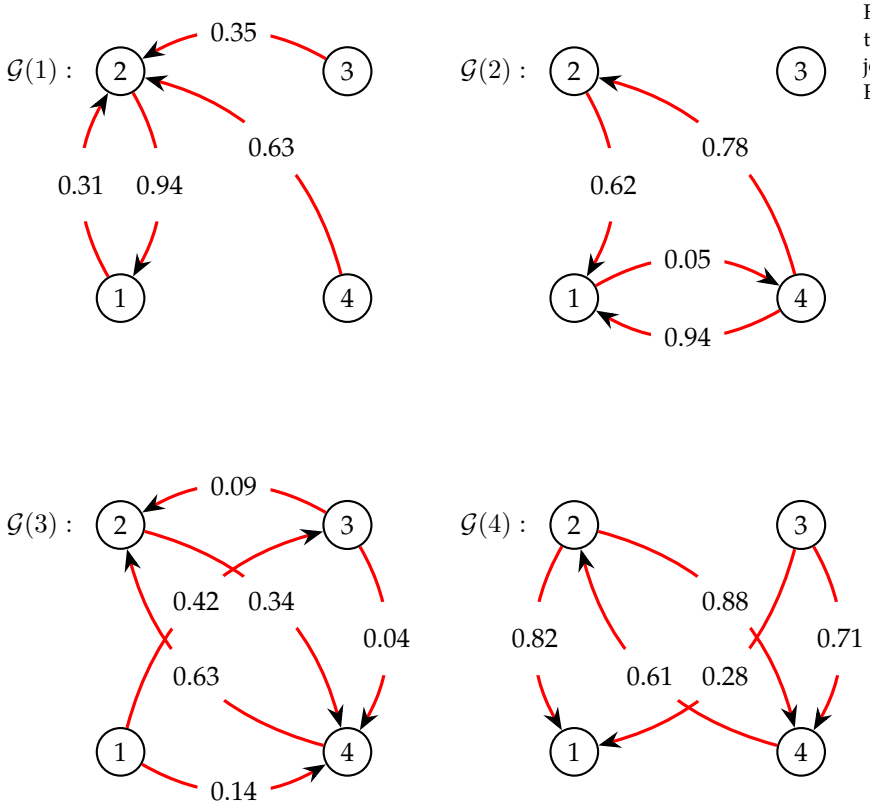


Figure 2.13: First four underlying topologies belonging to the 4-repeatedly jointly strongly connected sequence of Example 4.

Discrete-time

The formalism of difference equations is used in most literature since, in general, communication between agents occurs at discrete instants of time. A common discrete-time consensus protocol (see [40, 70]) is, $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$,

$$x_i(k+1) - x_i(k) = -\sum_{\substack{j=1 \\ j \neq i}}^{n} w_{ij}(k)(x_i(k) - x_j(k)) \tag{2.12}$$

where $\forall i \in \mathcal{N},\ \forall j \in \mathcal{N},\ \forall k \in \mathbb{N}_0$, $w_{ij}(k) \in \mathbb{R}_{\geq 0}$ and $w_{ij}(k) > 0$ if and only if $j \in \mathcal{N}_i(k)$. Thus, (2.12) can be rewritten as

$$\forall k \in \mathbb{N}_0,\ x_i(k+1) = w_{ii}(k)x_i(k) + \sum_{j \in \mathcal{N}_i(k)} w_{ij}(k)x_j(k), \tag{2.13}$$

where $w_{ii}(k) = 1 - \sum_{j \in \mathcal{N}_i(k)} w_{ij}(k)$. Clearly, $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$,

$$\sum_{j=1}^{n} w_{ij}(k) = 1. \tag{2.14}$$

In the following, we will assume that $w_{ii}(k)$ are positive. Let, $\forall k \in \mathbb{N}_0$, $\boldsymbol{x}(k) \in \mathbb{R}^n$ stack all information states, i.e.,

$$\forall i \in \mathcal{N},\ [\boldsymbol{x}(k)]_i = x_i(k). \tag{2.15}$$

Dynamics (2.12) can be rewritten in the corresponding matrix form as

$$\forall k \in \mathbb{N}_0,\ \boldsymbol{x}(k+1) = D(k)\boldsymbol{x}(k), \tag{2.16}$$

where, $\forall k \in \mathbb{N}_0$, $D(k)$ is a nonnegative and row-stochastic square matrix of order n with positive diagonal. For every $k \in \mathbb{N}_0$, by Definition 6, the weighted directed graph $\Gamma(D(k)) = (\mathcal{N}, \mathcal{A}(k), \mathcal{W}(k))$ has adjacency matrix $D(k)$, where $\forall i, j \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$,

$$[\mathcal{W}(k)]_{ij} = \begin{cases} w_{ij}(k) \in \mathbb{R}_{>0} & \text{if } j \in \mathcal{N}_i(k) \cup \{i\} \\ 0 & \text{otherwise} \end{cases}.$$

As a consequence,

$$\mathfrak{G}_w := \{\mathcal{G}_w(k)\}_{k \in \mathbb{N}_0} = \{(\mathcal{N}, \mathcal{A}(k), \mathcal{W}(k))\}_{k \in \mathbb{N}_0}$$

is the sequence of time-varying weighted directed graphs with adjacency matrices $D(k)$, $k \in \mathbb{N}_0$. Let's first analyze the case of a time-invariant topology, i.e., $\forall k \in \mathbb{N}_0$, $D(k) = D$, which leads to having system

$$\forall k \in \mathbb{N}_0,\ \boldsymbol{x}(k+1) = D\boldsymbol{x}(k). \tag{2.17}$$

By the latter,

$$\lim_{k \to \infty} \boldsymbol{x}(k) = \lim_{k \to \infty} D^k \boldsymbol{x}(0). \tag{2.18}$$

Whether system (2.17) converges to consensus or not depends on the spectral properties of matrix D. Let $\mathcal{G} = (\mathcal{N}, \mathcal{A}, \mathcal{W})$ with adjacency matrix D be the underlying network topology, i.e., a weighted directed graph of matrix D.

Proposition 7. *If the underlying network topology is strongly connected, system* (2.17) *achieves consensus, i.e.,*

$$\lim_{k\to\infty} \boldsymbol{x}(k) = \mathbf{1}_n x^*.$$

Proof. By Proposition 1, D is irreducible. By the latter and by Corollary 1, since D has a positive diagonal, D is primitive. Because of (2.14), D is row-stochastic, hence

$$D\mathbf{1}_n = \mathbf{1}_n$$

and 1 is an eigenvalue of D with corresponding right eigenvector $\mathbf{1}_n$. Because of Theorem 3, all eigenvalues of D lie within discs centered in w_{ii} and with radius $1 - w_{ii}$.[5] Finally, because of Theorem 2, $\rho(D) = 1$ is the spectral radius of D and, by Theorem 2.6,

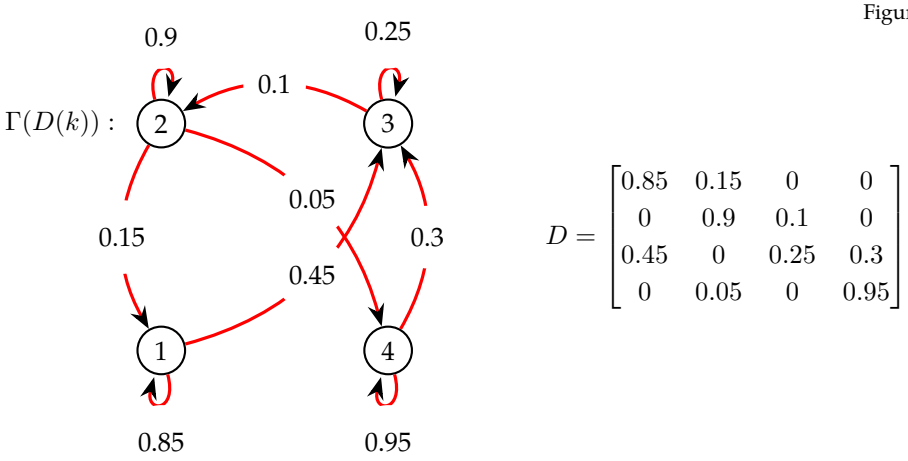
$$\lim_{k\to\infty} \boldsymbol{x}(k) = \lim_{k\to\infty} D^k \boldsymbol{x}(0) = \lim_{k\to\infty} (\rho(D)^{-1} D)^k \boldsymbol{x}(0) = \mathbf{1}_n \boldsymbol{w}_n' \boldsymbol{x}(0),$$

where $\boldsymbol{w}_n'$ is the left eigenvector[6] of D with eigenvalue 1. For $x^* := \boldsymbol{w}_n' \boldsymbol{x}(0)$, the proof is completed. □

[5] Therefore, no eigenvalues are strictly larger than 1.

[6] By Theorem 2.4, $\boldsymbol{w}_n' \mathbf{1}_n = 1$.

Example 5. *The network is composed of* 4 *agents with the same initial information states and underlying network topology as Example 1. Incoming weights to each node are normalized, such that* (2.14) *holds. This is illustrated in Figure 2.14. A numerical simulation is run until* $k = 50$*, so that*



$$D = \begin{bmatrix} 0.85 & 0.15 & 0 & 0 \\ 0 & 0.9 & 0.1 & 0 \\ 0.45 & 0 & 0.25 & 0.3 \\ 0 & 0.05 & 0 & 0.95 \end{bmatrix}$$

Figure 2.14: $\Gamma(D(k))$ for Example 5.

agents asymptotically achieve an agreement at $x^* = 0.92$*. The outcome is presented in Figure 2.15.*

As for the continuous case, a stricter condition for consensus is that the underlying network topology contains a directed rooted spanning tree.

Proposition 8 ([89, Theorem 2.20])**.** *System* (2.17) *achieves consensus if and only if the underlying network topology contains a rooted directed spanning tree.*

Proof. See [89, Theorem 2.20]. □

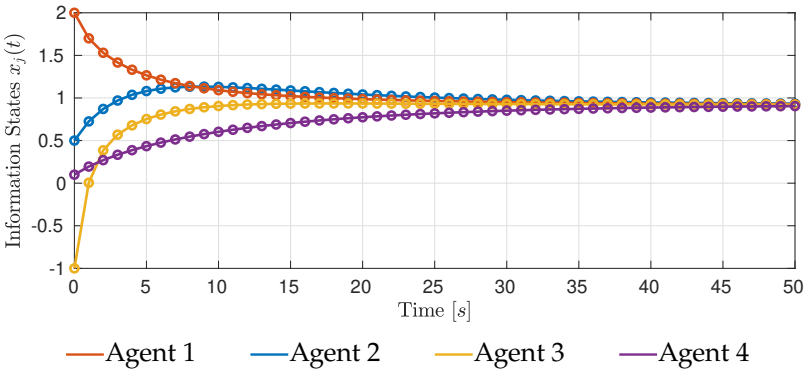


Figure 2.15: Discrete-time numerical simulation for Example 5.

However, as in Remark 1, we do not consider this condition to be meaningful for the problem at hand, since it does not allow agents to cooperate.

Example 6. *The topology of Example 5 is slightly changed, so that Agent 4 does not have any parent. Thus, the topology is not strongly connected, but contains a directed rooted spanning tree (with root in 4). As shown in Example 2, there is no cooperation, but all agents will eventually reach asymptotically the initial information state of the root agent. Topology is in Figure 2.16 and numerical simulations are presented in Figure 2.17.*

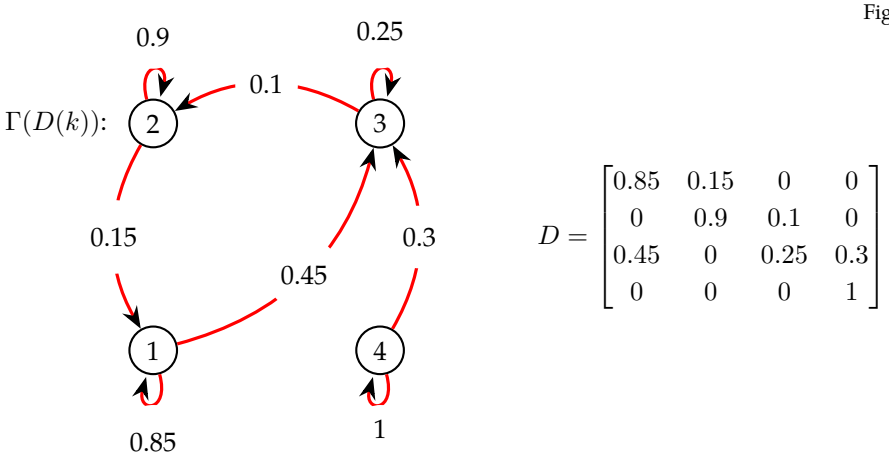


Figure 2.16: $\Gamma(D(k))$ for Example 6.

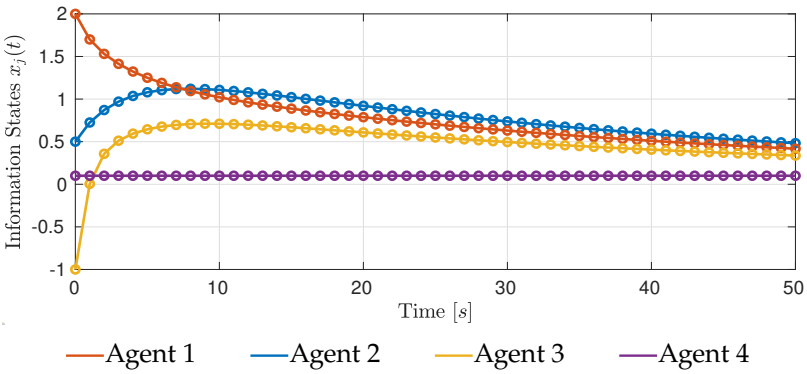


Figure 2.17: Discrete-time numerical simulation for Example 6. The simulation is stopped after 50 iterations; however, it shows that agents asymptotically converge towards the initial information state of Agent 4.

For the time-variant case, for which $w_{ij}(k)$ are not constant through time, the results for the discrete-time case are analogous to the ones for the continuous-time case.

Proposition 9 ([89, Theorem 2.39])**.** *System (2.16) achieves consensus if the sequence of underlying network topologies is repeatedly jointly strongly connected.*

Proof. The proof follows directly from [89, Theorem 2.39]. □

Example 7. *As for Example 3, a sequence of randomly generated strongly connected weighted digraphs, say $\mathfrak{G}$, is employed. The network is composed of 4 agents with the same initial information states as Example 1. Results are illustrated in Figure 2.18.*

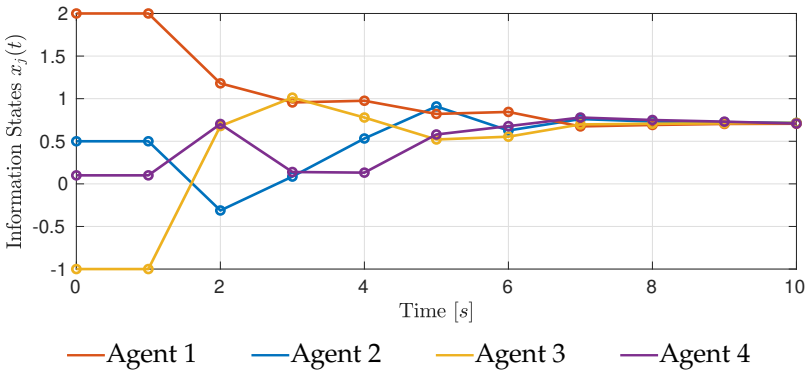


Figure 2.18: Asymptotic convergence to the consensus for Example 7.

The scenario of Example 7 can be extended by considering a more general condition on the sequence of topologies, as in Proposition 9.

Example 8. *Under the restriction of 4-repeatedly joint strongly connected topology and $\sum_{\in \mathcal{N}_i} w_{ij}(k) < 1$, a sequence of weighted directed graph $\mathfrak{G}$ is randomly generated. The first 8 elements of such a sequence are in Figure 2.20, is repeatedly jointly strongly connected. Simulation results are in Figure 2.19.*

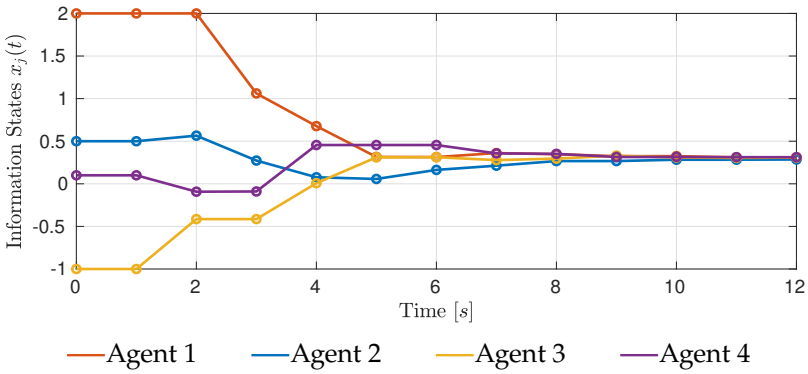


Figure 2.19: Discrete-time numerical simulation for Example 8.

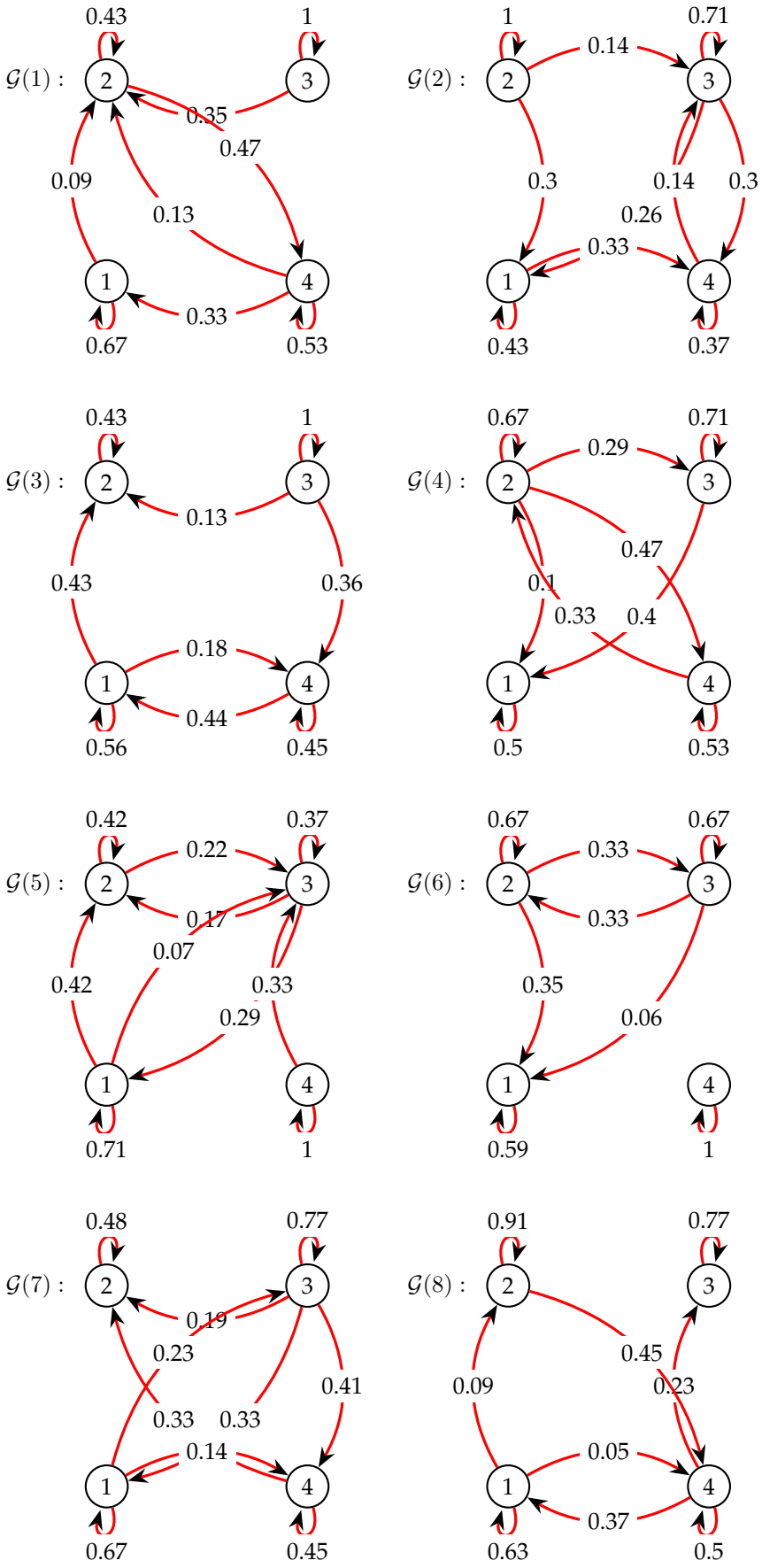


Figure 2.20: Underlying network topology for Example 8.

2.3.2 *Max-consensus*

Many distributed applications require that agents reach an agreement on the maximum information state, e.g., when agents have to distributively elect a leader. As for the previous section, each agent updates its information state according to the pieces of information received from neighbors. Contrarily to what done for the average-consensus, max-consensus protocols have been presented only in discrete-time frameworks. In fact, the most employed max-consensus protocol is as follows:

$$\forall i \in \mathcal{N},\ \forall t \in \mathbb{R}_{\geq 0},\ x_i(k+1) = \max_{j \in \mathcal{N}_i(k) \cup \{i\}} x_j(k). \tag{2.19}$$

For the case of a time-invariant underlying network topology, we refer to the contribution given by [75]. In case the topology is time-variant, [76] provides a valid analysis. The main results of both contributions are analyzed in what follows. We initially review a graphical theoretical concept.

Definition 11. *The length of a path in a graph is the number of arcs in that path.*

Definition 12 (Longest minimum path)**.** *In a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, the minimum path from node $i \in \mathcal{N}$ to a node $j \in \mathcal{N}$ is the shortest existing path from i to j. The* longest minimum path *of the graph is the maximum among all minimum paths' lengths. Formally, this is*

$$\ell(\mathcal{G}) = \max_{i,j \in \mathcal{N}} \{|i,j|_{min}\},$$

where, $\forall i, j \in \mathcal{N}$, $|i,j|_{min}$ is the length of the minimum path from i to j (minimum paths and longest minimum paths are nonunique).

Example 9. *A network composed of $n = 6$ nodes is given in Figure 2.21. The longest minimum path is coloured in blue.*

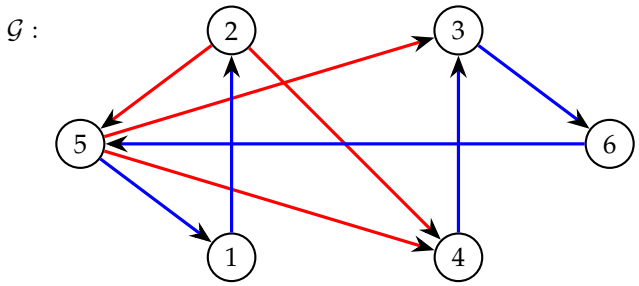


Figure 2.21: Highlighting the minimum path for the network of Example 9, which starts in Node 4 and finishes in Node 2.

The following definition differentiates this case from (2.7).

Definition 13. *Agents in set $\mathcal{N}$ achieve max-consensus if*

$$\exists \bar{k} \in \mathbb{N}_0 :\ \forall k \geq \bar{k},\ \forall i \in \mathcal{N},\ x_i(k) = \bar{x} := \max_{j \in \mathcal{N}} x_j(0). \tag{2.20}$$

The following result applies to time-invariant underlying network topologies.

Proposition 10 ([75, Corollary 4.2]). *Agents in set $\mathcal{N}$ iterating (2.19) over a strongly connected and time-invariant topology $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ achieve max-consensus in at most $\ell(\mathcal{G})$ iterations.*

Proof. See [75]. □

Example 10. *An underlying network topology as the one in Figure 2.21 is given. Its adjacency matrix is*

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

The vector of initial information states is

$$\boldsymbol{x}(k) = [1, 2, -1, 4, 3, 0]'.$$

Numerical results are in Figure 2.22.

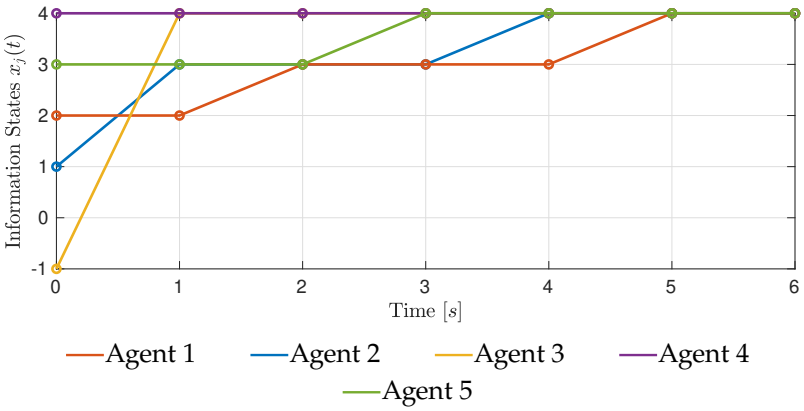


Figure 2.22: Discrete-time numerical simulation for Example 10.

The following result generalizes the findings to a time-variant underlying network topology.

Proposition 11 ([76, Theorem 4.3]). *Agents in set $\mathcal{N}$ iterating (2.19) over a sequence of time-varying strongly connected topologies achieve max-consensus within at most $n - 1$ iterations.*

Proof. See [76]. □

Example 11. *A sequence of strongly connected topologies is randomly generated and given in Figure 2.24. Agents, whose initial information states are as in Example 10, iterate the max-consensus protocol (2.19) and achieve an agreement to the maximum value in $n - 1 = 5$ iterations.*

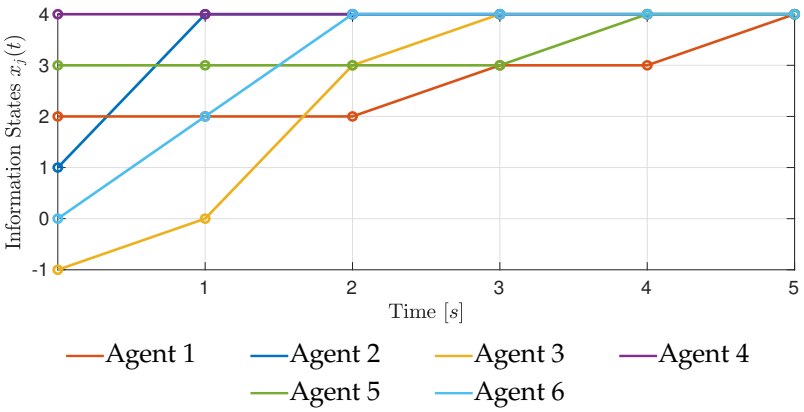


Figure 2.23: Discrete-time numerical simulation for Example 11.

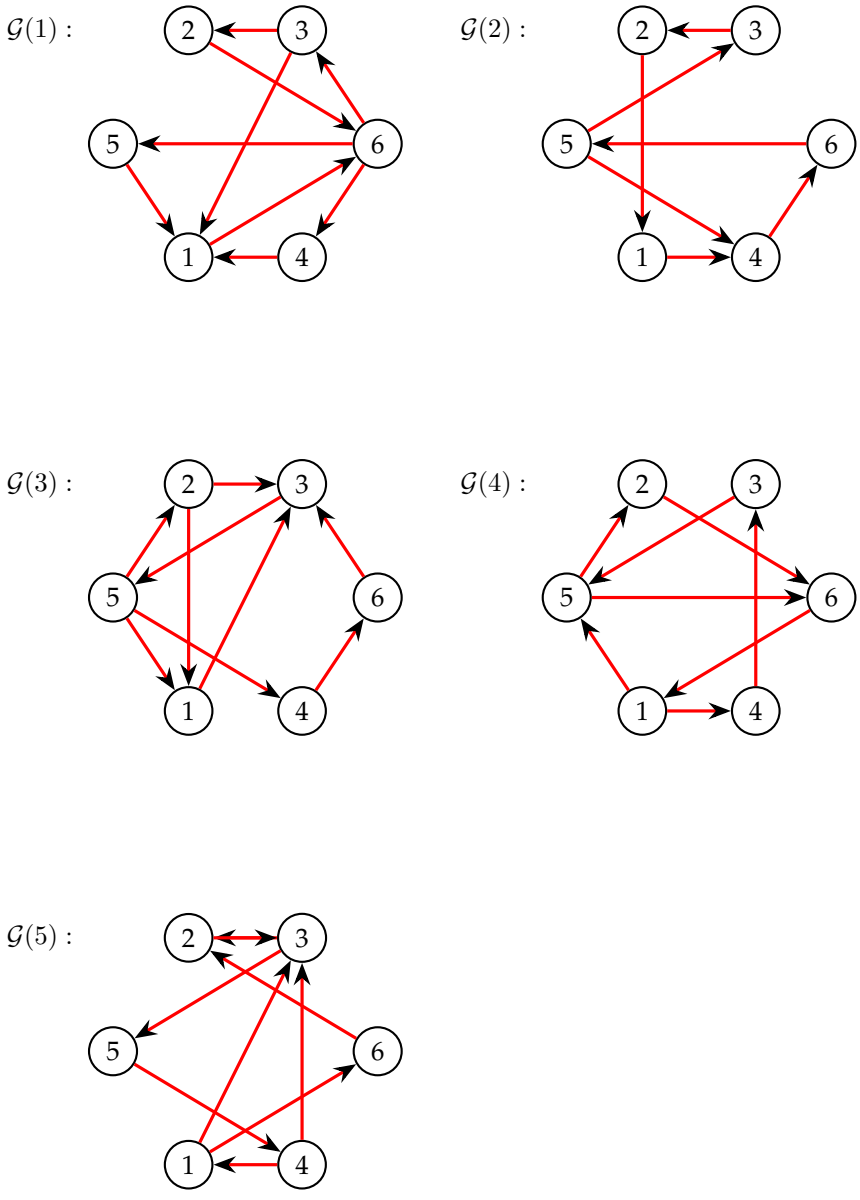


Figure 2.24: Underlying network topology for Example 11.

3
Using Consensus for Practical Problems

Consensus is employed in scenarios in which agents need to distributively achieve an agreement on a quantity of common interest. Among possible practical applications, control of vehicular traffic has attracted much interest due to the social and economical impact, as well as to the ease of implementation and verification. Vehicular traffic is a heterogeneous multi-agent system in which agents have different objectives and need to get to an agreement in order to homogenize the flow, thus decreasing hazards and increasing throughput, see, e.g., [116]. The following two scenarios describe this belief:

(i) Vehicles in a highway have different desired cruising speed. Pursuing their individual desired speeds would lead to traffic jam and hazards, see [69]. Also, when they wish to change lane, they should agree with all other traffic participants about the entry point.

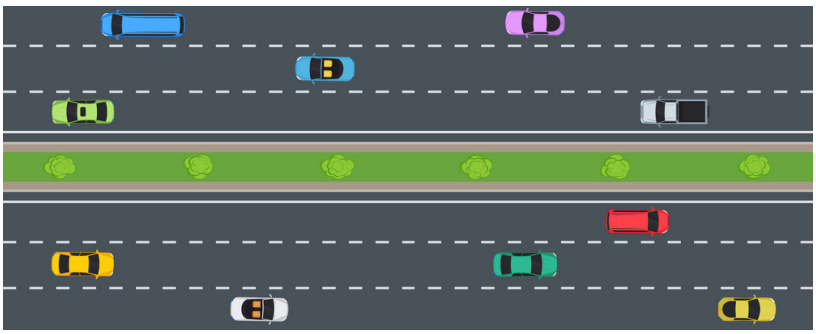


Figure 3.1: Highway.
Copyright by *Dreamstime.com*

(ii) Vehicles crossing an intersection need to agree on who passes first. Nowadays, traffic lights let the traffic reach an agreement on this.

In an autonomous vehicles scenario, agreement among all traffic participants is the milestone for any distributed control strategy. In fact, rather than employing a centralized omniscient controller, recent effort in automotive research have been made towards distributed control schemes, see, e.g., [56] and [43]. Indeed, adopting a distributed solution allows for real-time implementability, does not require high computational power, is compatible with plug-and-play scenarios in which vehicles enter and exit the environment, and its computational complexity does not increases with the number of vehicles. In contexts

Figure 3.2: Road intersection. Copyright by *Dreamstime.com*

where autonomous systems must achieve agreement on quantities of interest, consensus can be successfully employed. Vehicle to vehicle communication (V2V) is assumed, so that information can be shared between traffic participants. In what follows, Scenarios (i)-(ii) are analyzed in the following sections and the content of some scientific publication of the candidate is extensively analyzed and illustrated.

3.1 Consensus for Automating Road Intersections

Partial results of the presented section have been published in Molinari and Raisch [59] and Molinari et al. [65]. Partial results of the presented section have also been published in [67]. The candidate is the first author of all these contributions. The second author, Dr.-Ing. Alexander Katriniok, has contributed to the parts regarding Model Predictive Control and Simulation Results.

One big challenge for automating vehicular traffic in urban scenarios is the automation of road intersections, which tries to address the question of how to let vehicles coming from different direction cross while (i) avoiding collisions with others and (ii) maximizing the throughput. These two goal have been already analyzed in research. In fact, [18] proposes a distributed Model Predictive Control (MPC) scheme that determines vehicles' crossing order by solving two convex quadratic programs. Heuristics for determining the crossing order can be found in [19] and they pledge lower complexity an savalbility (in exchange for suboptimality). However, [19], as well as [39], require a central decision making for the crossing order, thus resulting in a non-fully distributed approach. The candidate has presented in [59] a fully distributed control scheme in which vehicles distributively negotiate their crossing order. This has been possible thanks to a Consensus-Based Auction Algorithm (CBAA-M) that let vehicles distributively agree on the crossing order. However, these contributions do not address real-time implementability. This topic is analyzed in [41], where a real-time capable distributed nonlinear MPC is designed. Each vehicle avoids collisions with vehicles having higher priority, and the crossing order is decided by the distributed MPCs. However, priorities are fixed and assigned in a centralized fashion.

Fully distributed and *real-time capable* are two paramount properties for AV's control schemes. The topic of this section is producing such a control scheme and showing that consensus is a milestone on which such an approach is built. In fact, the underlying idea is to bring together the CBAA-M algorithm by [59][1] for negotiating priorities and the real-time capable distributed nonlinear MPC scheme by [41].

[1] This algorithm has been originally designed by the candidate in 2018, see Molinari and Raisch [59].

3.1.1 Problem description

The distributed control scheme relies on V2V communication. Each vehicle (or agent), after solving an optimal control problem for its own trajectory, shares the result with other agents. The following assumptions, reasonable for an in-vehicle implementation, are hold:

Assumption 2. *A1. Only single intersection scenarios are considered; A2. A single lane is available per direction; A3. The desired route of every agent together with its desired speed are determined by a high-level route planning algorithm; A4. Agents are equipped with V2V communication; A5. No communication failures or package dropouts occur; A6. The MPC solutions at time k are available to all agents at time $k+1$; A7. Vehicle states are measurable and not subject to uncertainty.*

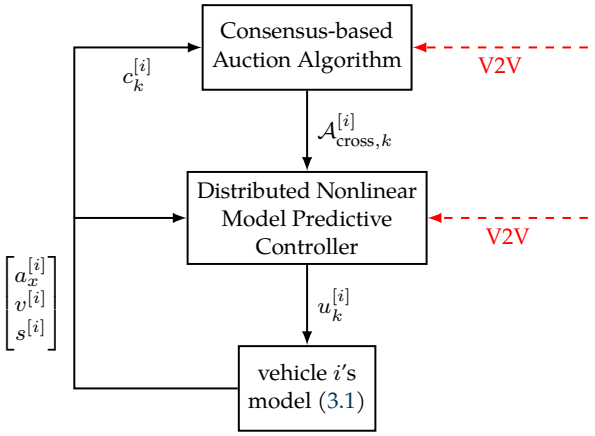


Figure 3.3: Hierarchical control structure.

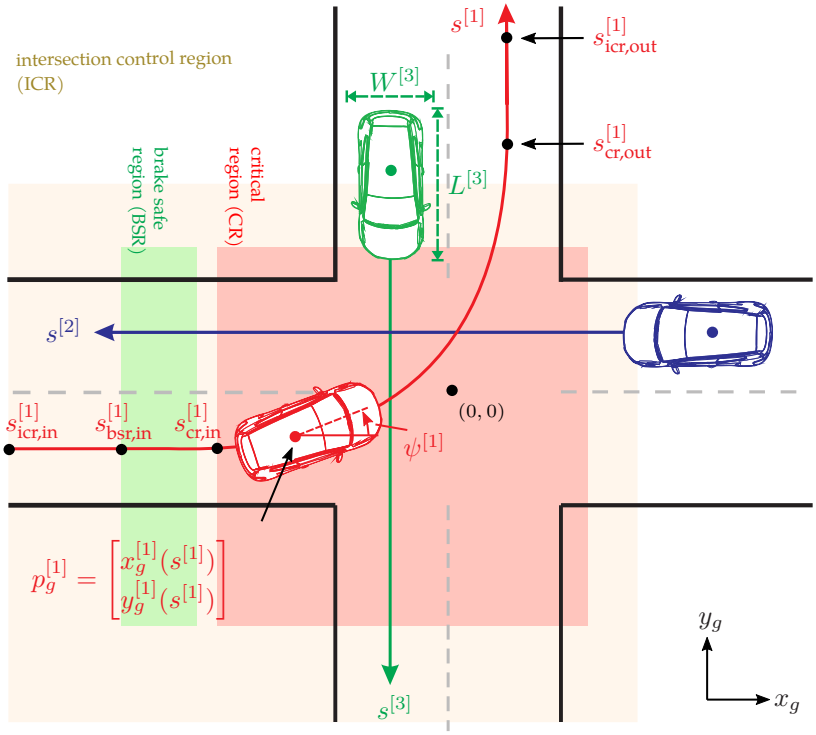


Figure 3.4: Intersection in global coordinates with origin at $(0, 0)$. Regions of the intersection are shown for Agent 1: Intersection Control Region (ICR, blue), Brake Safe Region (BSR, green), Critical Region (CR, red), and outside ICR (white).

When two agents enforce collision avoidance (CA) constraints towards each other, there is the need of solving a dual optimization problem, which carries extra complexity. A common approach to prevent the need of a dual optimization problem is to let only one agent of the pair enforce a CA constraint towards the other. In [41], agents are given fixed a-priori priorities, so that each agent holds CA constraints only towards those with higher priority. However, this scheme assigns priorities based on initial conditions and in a centralized fashion and does not account for the current traffic state. To tackle the online assignment of priorities, the hierarchical control scheme in Fig. 3.3 is designed. By interacting with other traffic participant, each agent runs *CBAA-M* thus obtaining a set of higher priority vehicles. The nonlinear model predictive controller holds CA constraints towards higher priority vehicles, thus deciding the crossing order, and therefore the

input to each vehicle's kinematics. This scheme shows how consensus is employed in practical problems. It is also clear that an increased efficiency of consensus (under the point of view of converging time and employed resources) is of extreme interest in many practical applications.

3.1.1.1 Kinematic Agent Model

The set of agents involved in the motion planning problem is $\mathcal{N} := \{1, \dots, N_A\}$ where $N_A \in \mathbb{N}$. Each agent $i \in \mathcal{A}$ is assumed to move along an *a priori* known path (Assumption A3), parameterized by its path coordinate $s^{[i]}$, see 3.4. For such kind of coordination problems, a simplified kinematic modeling approach is common in literature, see, e.g, [38, 59]. In this work, the time evolution of the agent's velocity $v^{[i]}$ and path coordinate $s^{[i]}$ are described by a double integrator model whereas drivetrain dynamics is modeled by a first-order lag element and acceleration is denoted by $a_x^{[i]}$. This leads to the following linear time-invariant state space model

$$\frac{d}{dt}\begin{bmatrix} a_x^{[i]} \\ v^{[i]} \\ s^{[i]} \end{bmatrix} = \underbrace{\begin{bmatrix} -1/T_{a_x}^{[i]} & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}}_{A^{[i]}} \underbrace{\begin{bmatrix} a_x^{[i]} \\ v^{[i]} \\ s^{[i]} \end{bmatrix}}_{\mathbf{x}^{[i]}} + \underbrace{\begin{bmatrix} 1/T_{a_x}^{[i]} \\ 0 \\ 0 \end{bmatrix}}_{B^{[i]}} \underbrace{a_{x,\text{ref}}^{[i]}}_{u^{[i]}}, \tag{3.1}$$

where $T_{a_x}^{[i]}$ stands for the dynamic drivetrain time constant and $u^{[i]} = a_{x,\text{ref}}^{[i]}$ is the reference acceleration (sent to the actuator). System (3.1) is discretized using a zero-order hold, thus yielding

$$\mathbf{x}_{k+1}^{[i]} = A_d^{[i]} \mathbf{x}_k^{[i]} + B_d^{[i]} u_k^{[i]}, \tag{3.2}$$

where $A_d^{[i]} = e^{A^{[i]} T_s}$, $B_d^{[i]} = \int_0^{T_s} e^{A^{[i]}(T_s - \tau)} B^{[i]} \mathrm{d}\tau$, and

$$\mathbf{x}_k^{[i]} := [a_{x,k}^{[i]},\ v_k^{[i]},\ s_k^{[i]}]',$$

with the subscript k denoting that the variable is sampled at time kT_s. While the kinematic agent model (3.2) describes Agent i's motion along its local path coordinate $s^{[i]}$, the respective global coordinates $(x_g^{[i]}(s^{[i]}), y_g^{[i]}(s^{[i]}))$ are used to formulate CA constraints. The map $\mathcal{F}_p^{[i]} : s^{[i]} \mapsto (x_g^{[i]}, y_g^{[i]})$ relates the local path coordinate $s^{[i]}$ to the corresponding global Cartesian coordinates. The heading angle $\psi^{[i]}(s^{[i]})$ and the path curvature $\kappa^{[i]}(s^{[i]})$ are yielded by $\mathcal{F}_\psi^{[i]} : s^{[i]} \mapsto \psi^{[i]}$ and $\mathcal{F}_\kappa^{[i]} : s^{[i]} \mapsto \kappa^{[i]}$, respectively[2]. A thorough description of such function can be found in [41, Sec. IIb].

[2] They can be obtained either as separate spline curves or obtained from the first and second derivative of $\mathcal{F}_p^{[i]}$.

3.1.1.2 Intersection Model

The intersection is divided in regions as in [41, Sec. IIc], see Fig. 3.4. In the *intersection control region* (ICR), that is, for $s_{\text{icr,in}}^{[i]} \leq s^{[i]} < s_{\text{icr,out}}^{[i]}$, the controller needs to enforce collisions avoidance towards crossing agents and agents driving ahead in the same lane. When entering the *brake safe region* (BSR), defined by $s_{\text{bsr,in}}^{[i]} \leq s^{[i]} < s_{\text{bsr,out}}^{[i]}$, agents are still able to stop safely before the *critical region* (CR). Only in the CR,

i.e., $s^{[i]}_{\text{cr,in}} \leq s^{[i]} < s^{[i]}_{\text{cr,out}}$, collisions with crossing agents may happen. Once abandoned the CR, only rear-end collision avoidance needs to be enforced.

3.1.2 *Consensus for crossing priorities*

CBAA-M is a powerful algorithm based on consensus that allows vehicles (agents) to negotiate priorities in a fully distributed fashion. The candidate has first presented and named this algorithm in Molinari and Raisch [59], taking inspiration from the auction algorithm of [23].

At every sampling time k, agents in set $\{i \in \mathcal{N} \mid s^{[i]}_k \leq s^{[i]}_{\text{cr,out}}\}$ participate in a distributed auction and bid for having the highest possible priority. The biddable quantity is determined by agent's velocity and position. The underlying communication network topology at sampling time $k \in \mathbb{N}_0$ is modeled by the directed graph $(\mathcal{N}, \mathcal{A})$, where $\mathcal{A}$ is the set of arcs, i.e. $(i, j) \in \mathcal{A}$ iff at sampling time k vehicle i transmits information to vehicle j. The result of this algorithm is that, at every $k \in \mathbb{N}_0$, each vehicle $i \in \mathcal{N}$ obtains a set of higher priority vehicles, i.e., $\mathcal{A}^{[i]}_{\text{cross},k} \subset \mathcal{N}$, towards which it will enforce CA constraints.

3.1.2.1 *Bid computation*

Reasonably, faster approaching vehicles (or vehicles closer to the BSR) should obtain higher priorities than vehicles driving more slowly (or being further away from the BSR). Moreover, a vehicle already inside the BSR must have higher priority than vehicles still outside. Accordingly, each vehicle $i \in \mathcal{N}$ at every sampling instant $k \in \mathbb{N}_0$ determines its own bid as follows:

$$c^{[i]}_k := \begin{cases} \alpha_1 v^{[i]}_k + \dfrac{\alpha_2}{(s^{[i]}_{bsr,in} - s^{[i]}_k)} & \text{if } s^{[i]}_{bsr,in} - s^{[i]}_k > \alpha_4 \\ \alpha_3 (s^{[i]}_k - s^{[i]}_{bsr,in}) + \alpha_5 & \text{else} \end{cases},$$

where $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5 \in \mathbb{R}_{>0}$ are design parameters.

Remark 3. *$\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5 \in \mathbb{R}_{>0}$ are chosen such that vehicles inside the BSR have always larger bids than vehicles outside, as in Fig. 3.5.*

Assumption 3. *For all distinct pairs of vehicles $i, j \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $c^{[j]}_k \neq c^{[i]}_k$.*

3.1.2.2 *Consensus-based Auction Algorithm*

CBAA-M is composed of two subsequent phases, a local auction phase in which vehicles place their bids, and a cooperative phase in which vehicles agree on the auction result. Let $\kappa \in \mathbb{N}$ denote the algorithm iteration. Each vehicle $i \in \mathcal{N}$ has two vectors, i.e., $\mathbf{v}^{\kappa}_i \in \{0 \dots N_A\}^{N_A}$ (containing the sorted list of agents) and $\mathbf{w}^{\kappa}_i \in \mathbb{R}^{N_A}_{>0}$ (containing the sorted list of bids), both initialized as null-vectors of dimension N_A. In order to avoid confusion of time indices, for the analysis, we drop the time index k from $c^{[i]}_k$, thus focusing only on algorithm iteration

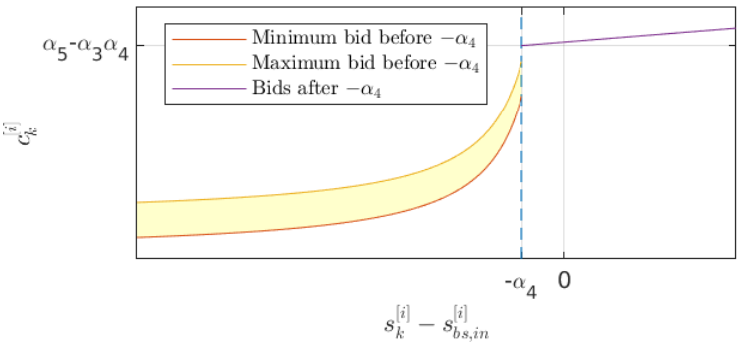


Figure 3.5: Bid as function of vehicle's distance to the BSR. For $s_{bsr,in}^{[i]} - s_k^{[i]} > \alpha_4$, the yellow area denotes all possible bids (since the bid is also a function of the speed). In this setting, $\alpha_1 = \frac{1}{10}$, $\alpha_2 = 5$, $\alpha_3 = \frac{1}{10}$, $\alpha_4 = 1$, $\alpha_5 = 7$.

Algorithm 1: (Phase 1). Local Auction: agents store locally their respective bid in the earliest possible position.

$\mathbf{v}_i^0 = \mathbf{0}_{N_A}$, $\mathbf{w}_i^0 = \mathbf{0}_{N_A}$
procedure BID($c^{[i]}$, $\mathbf{v}_i^{\kappa-1}$, $\mathbf{w}_i^{\kappa-1}$)
  $\mathbf{v}_i^\kappa \leftarrow \mathbf{v}_i^{\kappa-1}$
  $\mathbf{w}_i^\kappa \leftarrow \mathbf{w}_i^{\kappa-1}$
  loop:
  if $i \neq (\mathbf{v}_i^\kappa)_j, j = 1 \dots N_A$ **then**
    for $j = 1 \dots N_A$ **do**
      if $c^{[i]} > (\mathbf{w}_i^{\kappa-1})_j$ **then**
        $(\mathbf{v}_i^\kappa)_j \leftarrow i$
        $(\mathbf{w}_i^\kappa)_j \leftarrow c^{[i]}$
        close;
  close;

index κ.

Phase 1. Local Auction: at every iteration κ, each $i \in \mathcal{N}$ places, if its index is not already stored in $\mathbf{v}_i^\kappa$, its own bid $c^{[i]}$ in the earliest possible position of vector $\mathbf{w}_i^\kappa$. In the same position, it stores its index in vector $\mathbf{v}_i^\kappa$.

Phase 2. Consensus over the lists: after the local auction, each agent has its own version of $\mathbf{v}_i^\kappa$ and $\mathbf{w}_i^\kappa$. The network needs to agree on them. To this end, each agent $i \in \mathcal{N}$ transmits its vectors to agents in set $\mathcal{N}_i^{\text{out}} := \{j \in \mathcal{N} \mid (i,j) \in \mathcal{A}\}$ (namely, set of out-neighbors) and receives the vectors from agents in set $\mathcal{N}_i^{\text{in}} := \{j \in \mathcal{N} \mid (j,i) \in \mathcal{A}\}$ (namely, set of in-neighbors). Then, via a max-consensus protocol, it selects the best bid for each row of $\mathbf{w}_i^\kappa$ and puts in the same position of $\mathbf{v}_i^\kappa$ the respective agent's index. After terminating Phase 2, $(\mathbf{w}_i^\kappa)_j$ is the j-th highest bid that Agent i is aware of at iteration κ, and $(\mathbf{v}_i^\kappa)_j$ is the index of the agent having placed that bid.

Algorithm 2: (Phase 2). Consensus: agents agree on the auction result.

SEND $(\mathbf{v}_i^\kappa, \mathbf{w}_i^\kappa)$ to $j \in \mathcal{N}_i^{\text{out}}$
RECEIVE $(\mathbf{v}_h^\kappa, \mathbf{w}_h^\kappa)$ from $h \in \mathcal{N}_i^{\text{in}}$
procedure UPDATE($\{\mathbf{v}_h^\kappa\}_{h\in\mathcal{N}_i^{\text{IN}}\cup\{i\}}$, $\{\mathbf{w}_h^\kappa\}_{h\in\mathcal{N}_i^{\text{IN}}\cup\{i\}}$)
  for $j = 1 \dots N_A$ **do**
    if $\max_h((\mathbf{w}_h^\kappa)_j) > 0$ **then**
      $(\mathbf{a}_i^\kappa)_j \leftarrow \arg\max_h((\mathbf{w}_h^\kappa)_j)$
      $(\mathbf{v}_i^\kappa)_j \leftarrow (\mathbf{v}_{(\mathbf{a}_i^\kappa)_j}^\kappa)_j$
      $(\mathbf{w}_i^\kappa)_j \leftarrow \max_h((\mathbf{w}_h^\kappa)_j)$
    else
      close;
  close;

Proposition 12. *A multiagent system with a strongly connected network topology runs CBAA-M. Then,* $\exists \bar{\kappa} \in \mathbb{N}$*:*

$$\forall i, j \in \mathcal{N},\ \forall \kappa > \bar{\kappa}, \quad \mathbf{v}_i^\kappa = \mathbf{v}_j^\kappa = \mathbf{v}^\star = \text{argsort}(\mathbf{c}),$$
$$\mathbf{w}_i^\kappa = \mathbf{w}_j^\kappa = \mathbf{w}^\star = \text{sort}(\mathbf{c}),$$

where $\mathbf{c}_i = c^{[i]}$ *and* $\bar{\kappa} \leq N_A \ell$*, where* $\ell := \max_{i,j}(p_{ij})$*, and* p_{ij} *is the number of arcs in the shortest path*[3] *from* j *to* i*.*

[3] A path in $(\mathcal{N}, \mathcal{A})$ is a sequence of nodes, such that each pair of adjacent nodes is connected by a directed arc.

Proof. It is immediate to verify that, if $\exists \kappa_1$ such that, for one $i \in \mathcal{N}$, $\mathbf{v}_i^{\kappa_1} = \mathbf{v}^\star$ and $\mathbf{w}_i^{\kappa_1} = \mathbf{w}^\star$, then $\forall \kappa > \kappa_1$, $\mathbf{v}_i^\kappa = \mathbf{v}^\star$ and $\mathbf{w}_i^\kappa = \mathbf{w}^\star$.
Denote by $\iota_1 \in \mathcal{N}$ the first agent obtaining the solution, i.e. $\mathbf{v}_{\iota_1}^{\kappa_1} = \mathbf{v}^\star$ and $\mathbf{w}_{\iota_1}^{\kappa_1} = \mathbf{w}^\star$ for some $\kappa_1 \in \mathbb{N}_0$ (at the corresponding Phase 1). Agent ι_1 is clearly the first agent that stores, in the Phase 1 of iteration κ_1, the value $\min(\mathbf{c})$ in the last entry of vector $\mathbf{w}_{\iota_1}^{\kappa_1}$. Doing that, ι_1 stores also its own index in the last entry of $\mathbf{v}_{\iota_1}^{\kappa_1}$, thus proving $\iota_1 = \arg\min(\mathbf{c})$, namely ι_1 is the agent with the lowest bid. This is possible only if, at the end of Phase 2 of $\kappa - 1$, $\forall j = 1 \dots N_A - 1$, $(\mathbf{w}_{\iota_1}^{\kappa_1 - 1})_j = (\mathbf{w}^\star)_j$ (respectively, $(\mathbf{v}_{\iota_1}^{\kappa_1 - 1})_j = (\mathbf{v}^\star)_j$).
Let now $\iota_2 \in \mathcal{N}$ be the first agent such that

$$\forall j = 1 \dots N_A - 1, \qquad (\mathbf{w}_{\iota_2}^{\kappa_2})_j = (\mathbf{w}^\star)_j$$

(respectively, $(\mathbf{v}_{\iota_2}^{\kappa_2})_j = (\mathbf{v}^\star)_j$), for some $\kappa_2 \in \mathbb{N}_0$. As of above, it is also clear that $\iota_2 = \arg\min(\tilde{\mathbf{c}})$, where $\tilde{\mathbf{c}} := \{c \in \mathbf{c} \mid c > \min \mathbf{c}\}$, namely ι_2 is the agent with the second smallest bid.
By [75], since Phase 2 is based on max-consensus, propagating the first $(N_A - 1)$ entries of ι_2's vector starting at iteration κ_2 through the whole network (thus also to ι_1) requires at most $\ell \in \mathbb{N}$ iterations. This yields that $\kappa_1 \leq \kappa_2 + \ell$. Applying this recursively for every entry of the solution vectors, yields

$$\kappa_1 \leq \underbrace{\ell + \dots + \ell}_{N_A - 1},$$

equivalently, $\kappa_1 \leq (N_A - 1)\ell$. Propagating $\mathbf{v}^\star$ and $\mathbf{w}^\star$ to the network takes again at most ℓ iterations. This shows that $\bar{\kappa} \leq \kappa_1 + \ell \leq N_A \ell$, thus concluding the proof. □

3.1.3 *Distributed Motion Planning*

This section is extracted from Section 4 of the contribution "Real-Time Distributed Automation Of Road Intersections", submitted to the *2020 IFAC World Congress*, and derives from the collaboration with Dr.-Ing. Alexander Katriniok.

All vehicles' objectives (tracking and comfort) and some of their constraints (actuator constraints) are independent of other agents. Conversely, CA constraints couple different agent's optimal control problems (OCPs). As in [41], a primal decomposition technique is used to distribute the motion planning problem.

3.1.3.1 Separable Objectives

The local objectives of each agent, say $i \in \mathcal{N}$, are (i) to minimize the deviation of the agent's speed $v^{[i]}$ from the desired speed $v^{[i]}_{\text{ref}}$, and (ii) to ensure comfort onboard and driving efficiency by minimizing the acceleration $u^{[i]} = a^{[i]}_{x,\text{ref}}$. Summing these objectives along the prediction horizon (of length N) given time k yields

$$\sum_{j=1}^{N} \ell_j^{[i]}(\mathbf{x}^{[i]}_{k+j|k}, u^{[i]}_{k+j|k}) \tag{3.3}$$

where

$$\ell_j^{[i]}(\mathbf{x}^{[i]}_{k+j|k}, u^{[i]}_{k+j|k}) := q^{[i]} \left(v^{[i]}_{k+j|k} - v^{[i]}_{\text{ref},k+j|k}\right)^2 + r^{[i]} \left(u^{[i]}_{k+j|k}\right)^2 \tag{3.4}$$

and the terminal cost is

$$\ell_N^{[i]}(\mathbf{x}^{[i]}_{k+N|k}) := q_N^{[i]} \left(v^{[i]}_{k+N|k} - v^{[i]}_{\text{ref},k+N|k}\right)^2, \tag{3.5}$$

where $q^{[i]} > 0$, $q_N^{[i]} > 0$ and $r^{[i]} > 0$ are positive weights.

3.1.3.2 Separable Constraints

Due to actuator limitations, the demanded longitudinal acceleration is bounded, i.e.,

$$\forall j = 0, \dots, N-1,\ u^{[i]}_{k+j|k} \in \mathcal{U}^{[i]} := \left\{ u \in \mathbb{R} \mid \underline{a}^{[i]}_x \leq u \leq \overline{a}^{[i]}_x \right\}, \tag{3.6}$$

where $\overline{a}^{[i]}_x$ and $\underline{a}^{[i]}_x$ are upper and lower bounds. Vehicles do not drive backwards nor exceed the maximum speed, namely $\overline{v}^{[i]}$, thus

$$\forall j = 1, \dots, N,\ \mathbf{x}^{[i]}_{k+j|k} \in \mathcal{X}^{[i]}_{k+j|k} := \left\{ \mathbf{x} \in \mathbb{R}^3 \mid 0 \leq (\mathbf{x})_2 \leq \overline{v}^{[i]} \right\}. \tag{3.7}$$

Comfort and vehicle stability while turning can be guaranteed by constraining the lateral acceleration $a^{[i]}_y := \kappa^{[i]}(s^{[i]}){v^{[i]}}^2$, i.e.,

$$\forall j = 1, \dots, N,\ -\overline{a}^{[i]}_y \leq \kappa^{[i]}(s^{[i]}_{k+j|k}) \cdot (v^{[i]}_{k+j|k})^2 \leq \overline{a}^{[i]}_y, \tag{3.8}$$

where $\overline{a}^{[i]}_y$ is an appropriate upperbound. Staying within the friction circle (see, e.g., [86]) is important to the end of vehicle's stability. This can be enforced by constraining the total acceleration $\overline{a}^{[i]}_{\text{tot}}$ by

$$\begin{aligned} &\forall j = 1, \dots, N, \\ &\quad (a^{[i]}_{x,k+j|k})^2 + \left(\kappa^{[i]}(s^{[i]}_{k+j|k}) \cdot (v^{[i]}_{k+j|k})^2\right)^2 \leq (\overline{a}^{[i]}_{\text{tot}})^2. \end{aligned} \tag{3.9}$$

3.1.3.3 Coupling Constraints

To decouple CA constraints, only one vehicle per pair of possibly colliding vehicles needs to enforce the CA. To this end, we need to distinguish between collisions with crossing agents and rear-end collisions.

For each pair of **crossing agents**, say i and l, we leverage the output of CBAA-M. Set $\mathcal{A}^{[i]}_{\text{cross},k}$, resulting from CBAA-M, contains higher priority agents still inside of the CR. In contrast to [41], $\mathcal{A}^{[i]}_{\text{cross},k}$ is now time-varying instead of being *a priori* fixed.

To avoid **rear-end collisions**, only the following agent imposes CA constraints towards the preceding agent. For each agent $i \in \mathcal{N}$, at each sampling time k, the set $\mathcal{A}^{[i]}_{\text{ahead}} \subset \mathcal{A}$ defines agents that are, currently, in the same lane and ahead of Agent i.

Agent i at time k imposes CA constraints on vehicles, depending on the particular scenario:

1. Agent i inside of the ICR: CA constraints are imposed on agents
$$l \in \mathcal{A}^{[i]}_{c,k} := \mathcal{A}^{[i]}_{\text{cross},k} \cup \mathcal{A}^{[i]}_{\text{ahead}}. \tag{3.10}$$
2. Agent i outside of the ICR: only rear-end CA constraints are imposed on agents
$$l \in \mathcal{A}^{[i]}_{c,k} := \mathcal{A}^{[i]}_{\text{ahead}}. \tag{3.11}$$

All vehicles impose CA constraints towards current frontal vehicles. Additionally, vehicles inside of the ICR need to enforce CA constraints also towards higher priority vehicles.

Remark 4. *It can happen that, at sampling time k_0, one agent, say i, crosses earlier than a higher priority agent, say l. In any case, because of the way bids are computed, i will end up with a higher priority than l at the next step. In fact, at time $k > k_0$, $(s^{[i]}_k - s^{[i]}_{bs,in}) > (s^{[l]}_k - s^{[l]}_{bs,in})$, which implies, by the computation of bids for CBAA-M, that i ends up obtaining higher priority than l.*

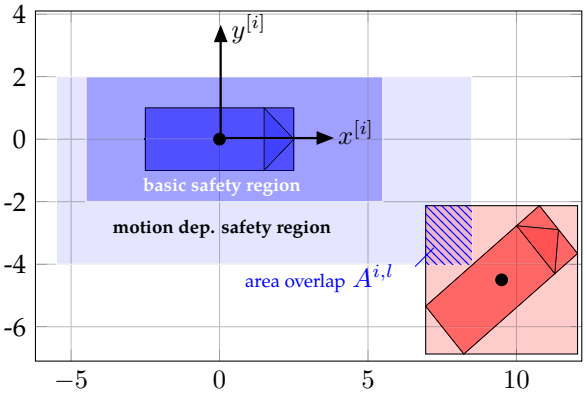


Figure 3.6: Agent i's safety region along with Agent l's bounding box in Agent i's Cartesian body frame.

For every agent l in the conflict set $\mathcal{A}^{[i]}_{c,k}$ at time k, we examine the area overlap of Agent i's safety region and Agent l's bounding box, namely $A^{i,l}$, see Fig. 3.6. Agent i's safety region is composed of a fixed *basic* safety region and a *motion dependent* safety region which depends on the relative motion with respect to Agent l. Collision avoidance is ensured if the overlap is zero. To this end, we introduce the equality constraint

$$A^{i,l}_{k+j|k} = \emptyset, \ \forall l \in \mathcal{A}^{[i]}_{c,k}. \tag{3.12}$$

for every time step $k + j$, $j = 1, \ldots, N$ over the prediction horizon. A thorough analysis of this latter constraint is in [41, Sec.IIIb].

3.1.3.4 Minimum Spatial Preview

To ensure collision avoidance, the spatial preview of every agent i, that is the lookahead in meters along the path coordinate $s^{[i]}$, has to be long enough. Results contained in [41] show that each agent, say i, has to leave the CR, at the latest, at the final time step $k+N$ of the prediction horizon, that is, $s^{[i]}_{k+N|k} \geq s^{[i]}_{\text{cr,out}}$. If this is not possible, then Agent i must stop before the stopping line, namely $s^{[i]}_{\text{stop}}$, before proceeding to the CR. This constraint can be expressed as

$$\left[-s^{[i]}_{k+N|k} + s^{[i]}_{\text{cr,out}}\right]_+ \cdot \left[s^{[i]}_{k+N|k} - s^{[i]}_{\text{stop}}\right]_+ = 0, \tag{3.13}$$

where $[x]_+ := \max\{0, x\}$.

3.1.3.5 Optimal Control Problem

By (Assumption A6), conflicting agents $l \in \mathcal{A}^{[i]}_{c,k}$ have transmitted at time $k-1$ their optimized position, velocity and heading trajectories, namely,

$$(x^{[l],\star}_{g,\cdot|k-1}, y^{[l],\star}_{g,\cdot|k-1}, \psi^{[l],\star}_{\cdot|k-1}, v^{[l],\star}_{\cdot|k-1}).$$

With this pieces of information at hand, every agent $i \in \mathcal{A}$ solves the following Optimal Control Problem (OCP) at time k

$$\min_{\{u_{k+j|k}\}_{j=0}^{N-1}} \ \ell^{[i]}_N(\mathbf{x}^{[i]}_{k+N|k}) + \sum_{j=0}^{N-1} \ell^{[i]}_j(\mathbf{x}^{[i]}_{k+j|k}, u^{[i]}_{k+j|k}) \tag{3.14a}$$

$$\text{s.t. } \mathbf{x}^{[i]}_{k+j+1|k} = A^{[i]}_d \mathbf{x}^{[i]}_{k+j|k} + B^{[i]}_d u^{[i]}_{k+j|k} \tag{3.14b}$$

$$u^{[i]}_{k+j|k} \in \mathcal{U}^{[i]}, \qquad j = 0, \dots, N-1 \tag{3.14c}$$

$$\mathbf{x}^{[i]}_{k+j|k} \in \mathcal{X}^{[i]}_{k+j|k}, \qquad j = 1, \dots, N \tag{3.14d}$$

$$-\overline{a}^{[i]}_y \leq a^{[i]}_{y,k+j|k} \leq \overline{a}^{[i]}_y, \quad j = 1, \dots, N \tag{3.14e}$$

$$(a^{[i]}_{\text{tot},k+j|k})^2 \leq (\overline{a}^{[i]}_{\text{tot}})^2, \quad j = 1, \dots, N \tag{3.14f}$$

$$A^{i,l}_{k+j|k} = \emptyset, \ \forall l \in \mathcal{A}^{[i]}_{c,k}, \quad j = 1, \dots, N \tag{3.14g}$$

$$\left[-s^{[i]}_{k+N|k} + s^{[i]}_{\text{cr,out}}\right]_+ \cdot \left[s^{[i]}_{k+N|k} - s^{[i]}_{\text{stop}}\right]_+ = 0, \tag{3.14h}$$

$$\mathbf{x}^{[i]}_{k|k} = \mathbf{x}^{[i]}_k, \tag{3.14i}$$

where,

$$\forall j = 1, \dots, N, \quad a^{[i]}_{\text{tot},k+j|k} = [(a^{[i]}_{x,k+j|k})^2 + (a^{[i]}_{y,k+j|k})^2]^{\frac{1}{2}}$$

is the total acceleration in (3.9) and

$$\forall j = 1, \dots, N, \quad a^{[i]}_{y,k+j|k} = \kappa^{[i]}(s^{[i]}_{k+j|k}) \cdot (v^{[i]}_{k+j|k})^2$$

the lateral acceleration in (3.8). At every time instant k, Agent i solves the OCP (3.14), thus yielding the sequence of optimal control inputs $(u^{[i],\star}_{k|k}, \dots, u^{[i],\star}_{k+N-1|k})$, whose first element, $u^{[i],\star}_{k|k}$, is executed. After optimization, the resulting optimized trajectories, namely,

$$(x^{[i]\star}_{g,\cdot|k}, y^{[i]\star}_{g,\cdot|k}, \psi^{[i]\star}_{\cdot|k}, v^{[i]\star}_{\cdot|k})$$

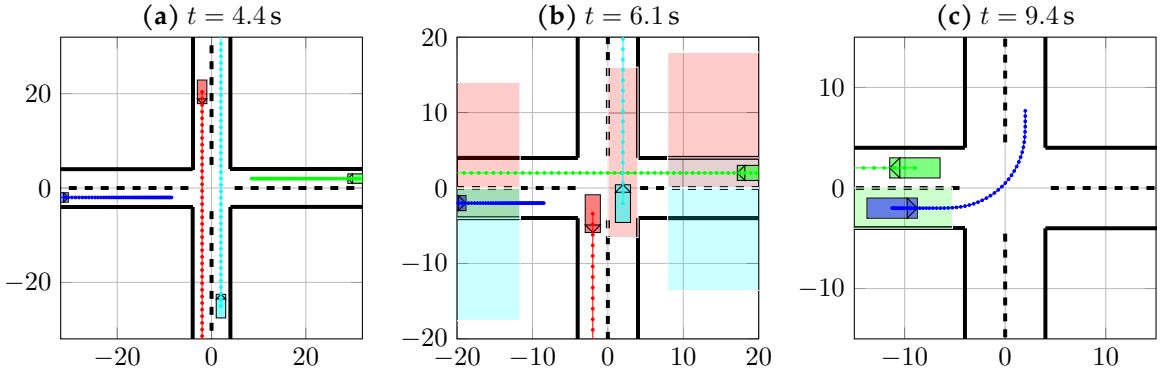


Figure 3.7: Snapshots Use Case 1: (Left) Agent 1 (red) and Agent 4 (cyan) cross first; (Middle) Agent 3 (green) proceeds after Agent 4 (cyan); (Right) Agent 2 (blue) turns left after Agent 3 (green) has left the CR. The middle and right figures show the safety region of each agent i in the color of the conflicting Agent l.

are transmitted to the other agents via V2V communication.

By the presence of (3.14g), and (3.14h), Problem (3.14) is nonconvex. By replacing equality and inequality constraints with penalty functions, see [77], it has been shown in [41, Sec. IV] that (3.14) can be recast in the form

$$\min_{u_{\cdot|k}^{[i]} \in U_k^{[i]}} \phi_k^{[i]}(u_{\cdot|k}^{[i]}; z_k^{[i]}), \tag{3.15}$$

so that it can be solved in real time using the proximal averaged Newton method for optimal control (PANOC) (see an extensive analysis in [96, 104]). To enforce constraint satisfaction, the quadratic penalty method, see [77, Chap. 17], is applied. In (3.15),

$$u_{\cdot|k}^{[i]} = [u_{k|k}^{[i]}, \dots, u_{k+N-1|k}^{[i]}]'$$

is the vector of predicted control actions of Agent i, and

$$z_k^{[i]} = [x_k^{[i],\top}, (x_{g,\cdot|k-1}^{[l],\top}, y_{g,\cdot|k-1}^{[l],\top}, \psi_{\cdot|k-1}^{[l],\top}, v_{\cdot|k-1}^{[l],\top})_{l \in \mathcal{A}_{c,k}^{[i]}}]'$$

is a parameter vector which provides to Agent i all necessary measured information. A detailed analysis of the method is contained in [41, Sec.IV].

3.1.4 *Simulation results*

This section is extracted from Section 4 of the contribution "Real-Time Distributed Automation Of Road Intersections", submitted to the *2020 IFAC World Congress*, and derives from the collaboration with Dr.-Ing. Alexander Katriniok.

A realistic intersection scenario with four agents as shown in Fig. 3.7a is considered. Agent 1 (red) crosses the intersection straight from North to South, Agent 2 (blue) approaches the intersection from the West and turns left while Agent 3 (green) and Agent 4 (cyan) crosses the intersection straight from East to West and South to North, respectively. Each agent has the same dimensions of $L^{[i]} = 5\,\mathrm{m}$ and

$W^{[i]} = 2\,\text{m}$ and the same drivetrain time constant of $T_{a_x}^{[i]} = 0.3\,\text{s}$. The initial positions in the global frame are: $(-2, 82)$ for Agent 1, $(-84, -2)$ for Agent 2, $(81, 2)$ for Agent 3 and $(2, -84)$ for Agent 4. Moreover, all agents have the same initial and reference velocity of $14\,\text{m/s}$ while the maximum speed is $15\,\text{m/s}$. Sampling time is $T_s = 0.1\,\text{s}$. MPC's prediction horizon consists of $N = 50$ steps. MPC's weights are chosen equally for every agent, that is, $q^{[i]} = q_N^{[i]} = 1$ and $r^{[i]} = 20$. Agents' safety region is parameterized as in [41]. Finally, we constrain the demanded longitudinal acceleration between -7 and $4\,\text{m/s}^2$ while the absolute lateral acceleration has to be less or equal to $3.5\,\text{m/s}^2$ and the total acceleration is bounded from above by $7\,\text{m/s}^2$. All simulations are run on Intel i7 machine at 2.9 GHz with Matlab R2018b while the nonlinear MPC controllers run in `C89` using the open source code generation tool `nmpc-codegen`, available at `github.com/kul-forbes/nmpc-codegen`.

To evaluate the interplay of the auction based algorithm and the distributed MPC control scheme, we investigate two use cases for the scenario outlined above:

1. Regular priority negotiation: the agents are negotiating priorities as in Section 3.1.2.
2. Emergency vehicle: Agent 2 is an emergency vehicle, requesting the highest priority at $t = 0.5\,\text{s}$.

Use Case 1: Regular Priority Negotiation

Figure 3.9 illustrates in each row i the optimized motion trajectories of Agent i; Figure 3.7 shows three snapshots of the maneuver in the global coordinate frame along with agents' safety regions. In addition, Figure 3.8 highlights agents' negotiated priorities until 8 s (when, after agents enter their respective BSR, priorities turn out to be constant). Initially, Agent 1 (red) exhibits the highest priority, followed by Agent 3 (green), Agent 4 (cyan) and Agent 4 (blue). By getting closer to the intersection, at $t = 2\,\text{s}$, Agent 1 (red) and Agent 4 (cyan), driving in the North/South direction, obtain the highest and second highest priority, respectively; Agent 2 (blue) and Agent 3 (green) are assigned

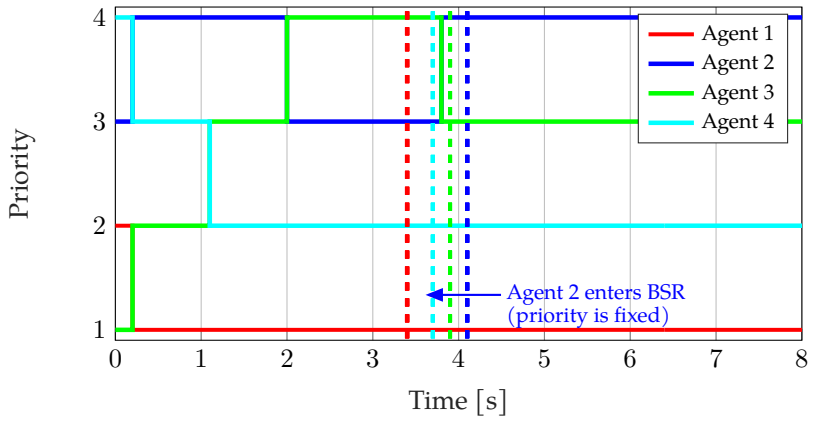


Figure 3.8: Use Case 1: Agent priorities are negotiated until the agents enter their BSR (indicated by dashed vertical line), then they remain constant.

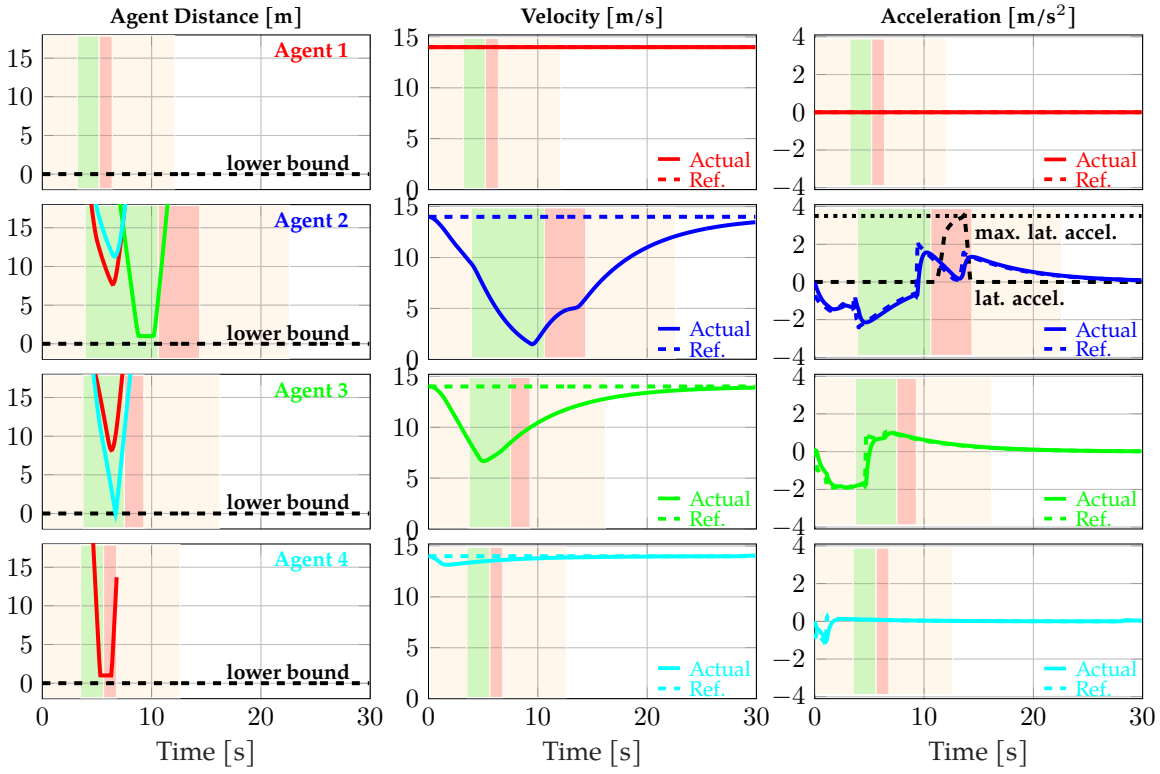


Figure 3.9: Use Case 1: From left to right in row i: (1) Distances between safety region and bounding box of agents i and l, (2) velocity and (3) acceleration of Agent i. The intersection regions are indicated by colored patches: inside ICR (beige), BSR (green), CR (red) and outside ICR (white).

the next higher priorities. At $t = 3.8\,\mathrm{s}$, Agent 3 (green) gets a higher priority than Agent 2 (blue), mainly because of Agent 2 (blue)'s deceleration before turning left. After $t = 4.1\,\mathrm{s}$, all agent priorities turn out to be fixed, since, inside of the CR, they are proportional only to the distance. Priorities also reflect in the average speeds. In fact, Agent 2 (blue) has both the lowest priority and the lowest average speed (same for other vehicles).

The optimized motion trajectories in Figure 3.9 prove that the time-varying (until $t = 4\,\mathrm{s}$) negotiated priorities do not cause discontinuities in acceleration or velocity. By comparing Fig. 3.7 and Fig. 3.8, one can observe that the crossing order is determined by the negotiated priority. In Use Case 2, we will show that this is not true in general. Initially, the acceleration of Agent 4 (cyan) is slightly negative; this is the case until it is assigned the 2$^{\mathrm{nd}}$ highest priority after $t = 1\,\mathrm{s}$. Agent 2 (blue) yields the way to Agent 3 (green), see Figure 3.7c, thus decelerating until before $t = 10\,\mathrm{s}$. On its way through the intersection, Agent 2 (blue) satisfies its lateral acceleration constraint (depicted in the third column). Concerning CA, the first column provides evidence that agent trajectories are safe as the distance between the agent's safety region and the other agent's bounding box is always greater or equal to zero. Moreover, velocities and accelerations are smooth and inside their designated bounds.

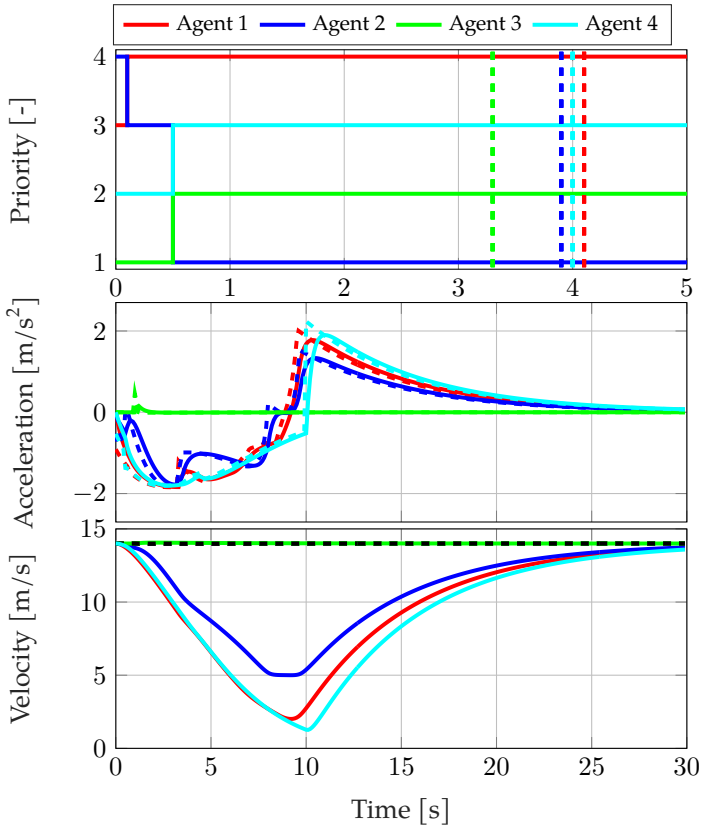


Figure 3.10: Use Case 2: Emergency vehicle switches from priority 3 to highest priority at $t = 0.5\,\mathrm{s}$.

Use Case 2: Emergency Vehicle

Initial conditions and scenario of the Use Case 2 are the same as Use Case 1, apart from Agent 2 (blue) — assumed to be an ambulance — receiving an emergency call at $t = 0.5\,\mathrm{s}$. The topic of this section is to investigate how a sudden change in priority affects the entire traffic. Because of the limited space, relevant trajectories and priorities are condensed in Figure 3.10. Until $t = 0.5\,\mathrm{s}$, Agent 2 (blue) retains the 3$^{\mathrm{rd}}$ negotiated priority. At $t = 0.5\,\mathrm{s}$, Agent 2 (blue) receives an emergency call, gets an arbitrarily high bid and negotiates the 1$^{\mathrm{st}}$ priority. Figure 3.10 confirms that acceleration and velocity do not show any discontinuities as priorities are changed suddenly. Agent 2 (blue) crosses the intersection without taking care of any agent. However, in contrast to Use Case 1, priorities do not determine the crossing order. In fact, Agent 3 (green) crosses first (the vertical lines in the priority plot indicate when the agents enter their BSR) without the need to decelerate. Clearly, by (3.10), Agent 3 (green) needs to enforce CA constraints inside of the intersection towards Agent 2 (blue). This is an advantage (and an additional *degree of freedom*) of our approach, with regards to, e.g., [65]. From the above discussion, it is evident that requirements in terms of comfort, performance, and safety are satisfied.

3.1.5 *Real-time capability*

Real-time capability depends on two distinct components: the convergence of CBAA-M to a solution and the solving time of the nonlinear MPC.
Concerning CBAA-M, Proposition 12 states that, at each sampling time, the network gets to the agreement in at most $N_A \ell$ iterations. We deal with state-of-the-art communication technology, i.e., 5G network used for automated driving. The technical specification [1][4] contains the newest global specifications for automated driving. Accordingly, for the scenario *Emergency trajectory alignment between UEs supporting V2X application*, the following performance aspects are defined:

- *reliability of communication links* is 99.999%;
- *max end-to-end latency* is 3 ms.

[4] 3GPP is a standards organization which develops protocols for mobile telephony.

Because of the high link reliability, it is straightforward to assume the underlying topology to be *fully-connected*, thus $\ell = 1$. Agreement is achieved in N_A iterations, thus

$$T_{\text{CBAAM}} \leq N_A \cdot 3 \text{ ms}.$$

For the case at hand, we assume indeed $T_{\text{CBAAM}} = 12$ ms.
Concerning the solving time of the nonlinear MPC, we refer to Fig. 3.11, which depicts agents' computation times for *Use Case 1*, simulated on a machine with the specifications in Section 3.1.4. T_{MPC} is at most 72 ms. Note that a dedicated setup would lead to an even faster convergence. However, also in a non-dedicated simulation environment, the overall execution time for the hierarchical controller is

$$T_{\text{CBAAM}} + T_{\text{MPC}} \leq 12 \text{ ms} + 72 \text{ ms} \leq 84 \text{ ms},$$

which is strictly less than the sampling time of 100 ms.

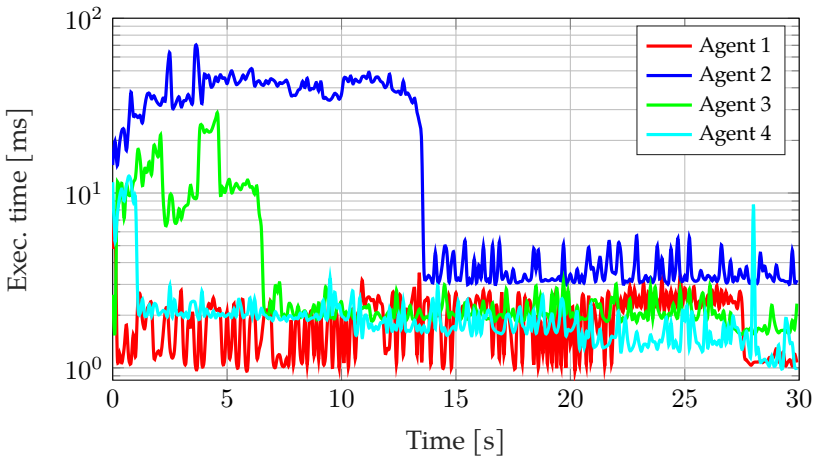


Figure 3.11: Use Case 1: Agent computation times.

3.2 *Automating Highways via Distributed Agreement*

The presented section is extracted from Molinari et al. [66]. The candidate is the first author of this contribution.

Research concerning autonomous vehicles is not only focusing on urban scenarios, but also on the study of extra-urban scenarios, see, e.g., [105]. A challenge in the control of automated[5] traffic in extra-urban scenarios is the control of a segment of a highway. This has been a research subject for a long time, see, e.g., [118], and various control structures have been proposed. One possible approach is to employ an all-knowing centralized controller that yields optimal trajectories tracking desired speeds, see e.g. [87]. However, [10] underlines that a centralized control strategy is practically not feasible. Besides lacking scalability and robustness, it requires an expensive vehicle-to-infrastructure (V2I) communication network.

[5] And semi-automated.

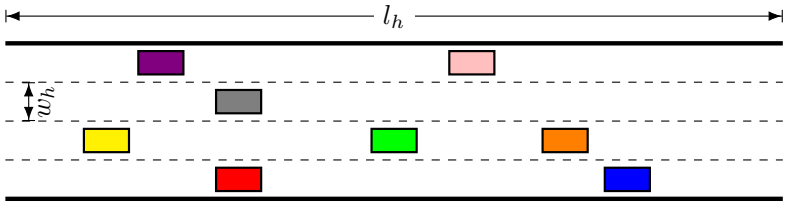


Figure 3.12: Highway.

As for the automation of road intersections in the urban scenario, also in this case a distributed approach would constitute a valid solution. Distributed control strategies do not require V2I communication (see, e.g., [52]). [58] proposes a cooperative control scheme for autonomous vehicles, which is generated by the decomposition of a single optimal problem into local problems, each associated with one vehicle. Many distributed control approaches (see, e.g., [11, 30, 81]) homogenize traffic, by forcing all vehicles to move at the same speed. However, the nature of traffic is heterogeneous: vehicles can have different desired speeds or depend on environmental conditions. As for the urban scenario, having a collection of autonomous agents with diverse goals motivates the need of employing consensus. In fact, consensus can be a valid solution in contexts where groups of autonomous agents have to agree or negotiate. Moreover, faster convergence and efficient usage of resources for consensus would contribute to reinforce this approach. Besides for the case of urban scenario (as extensively shown in the previous section), consensus has been already employed for the automation of extra-urban scenarios, see, e.g., [111], in which a consensus-based protocol is used for negotiating the merging point of vehicles accessing a highway ramp.

The control scheme proposed in this section preserves the heterogeneous nature of traffic while guaranteeing collision avoidance and preventing traffic congestion. After agreeing on a lane speed, each vehicle decides whether it wants to keep the lane or move into another one, faster or slower, based on its specific desired speed. Vehicles moving into a new lane distributively negotiate the entry point by using

the consensus-based auction algoithm introduced in the previous sections, i.e., CBAA-M. Finally, an onboard Model Predictive Controller (MPC) guarantees that collisions are avoided, while steering the vehicle into the desired lane.

3.2.1 *Problem Description*

We consider a stretch of a highway of length $l_h \in \mathbb{R}_{>0}$ with n parallel non-overlapping unidirectional lanes of identical width $w_h \in \mathbb{R}_{>0}$, as in Fig. 3.12. A set of vehicles (agents), denoted by $\mathcal{N} := \{1, \dots, m\}$, $m \in \mathbb{N}$, cruise on the highway.

The dynamics of each vehicle $i \in \mathcal{N}$ is described by a simple model (*bicycle model*, e.g. [83]), i.e., $\forall t \in \mathbb{R}_{\geq 0}$,

$$\begin{cases} \dot{x}_i(t) &= v_i(t)\cos(\vartheta_i(t)) \\ \dot{y}_i(t) &= v_i(t)\sin(\vartheta_i(t)) \\ \dot{\vartheta}_i(t) &= \frac{v_i(t)}{L}\tan(\beta_i(t)) \end{cases}, \tag{3.16}$$

where $x_i(t) \in \mathbb{R}_{\geq 0}$ and $y_i(t) \in \mathbb{R}_{\geq 0}$ are longitudinal and lateral positions of the vehicle's center of mass, $v_i(t)$ is its velocity, $\vartheta_i(t)$ the yaw angle, and $\beta_i(t)$ the steering angle. Velocity $v_i(t)$ and steering angle $\beta_i(t)$ are considered to be the control inputs[6]. L is the vehicles' length (for ease of notation, all vehicles are assumed to have the same length). Figure 3.13 illustrates the model parameters.

[6] A standard input variable for such models is often the acceleration. In cases where the drive-train dynamics is fast, we can guarantee a certain desired velocity in a short time.

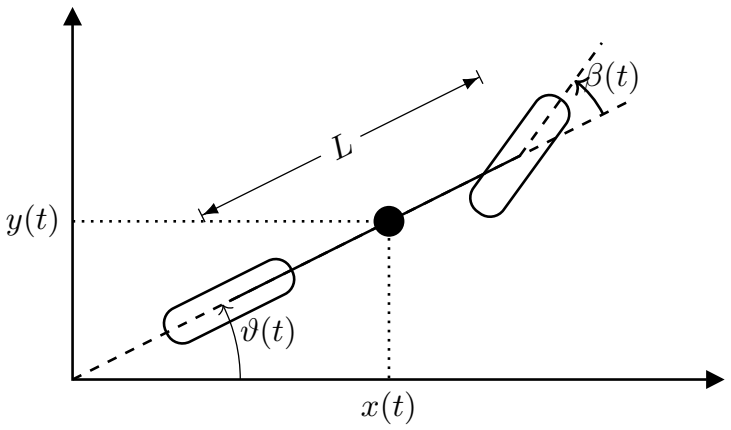


Figure 3.13: Bicycle model for the vehicle.

Throughout this section, $\boldsymbol{p}_i(t) := [x_i(t), y_i(t)]'$ denotes the position of vehicle $i \in \mathcal{N}$ on the highway, with $\boldsymbol{p}_i(t_0) := [x_{i_0}, y_{i_0}]'$ being its initial position. Each vehicle $i \in \mathcal{N}$ is given a a desired velocity $v_i^d \in [v_{\min}, v_{\max}]$.

Dealing with the presence of a digital controller (MPC), system (3.16) is sampled with sampling time $T_s \in \mathbb{R}_{>0}$. We denote

$$\forall k \in \mathbb{N}_0, \qquad x_i(k) := x_i(kT_s). \tag{3.17}$$

(similarly y_i, ϑ_i, v_i, and β_i). Let, $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $q_i(k)$ be the lane in which vehicle i is at time instant k. We say that a collision between vehicles $i, j \in \mathcal{N}$ occurs at the k-th sampling instant if the longitudinal

distance between i and j is below a given threshold (say $\tilde{D} > L$) and i and j are on the same lane, i.e.,

$$|x_i(k) - x_j(k)| < \tilde{D} \quad \wedge \quad q_i(k) = q_j(k). \tag{3.18}$$

As in the contribution from which this section is extracted, i.e., Molinari et al. [66], sampling time is chosen to be small enough so to ensure that collisions are avoided both at sampling instants and in between them. Clearly, the smaller is T_s, the more this assumption is reinforced. However, as carefully noted in [32], a small sampling time brings about poorer controller performance. In fact, given a fixed prediction horizon of the MPC, a smaller T_s implies more prediction steps to account for.

3.2.2 *Distributed Control Scheme*

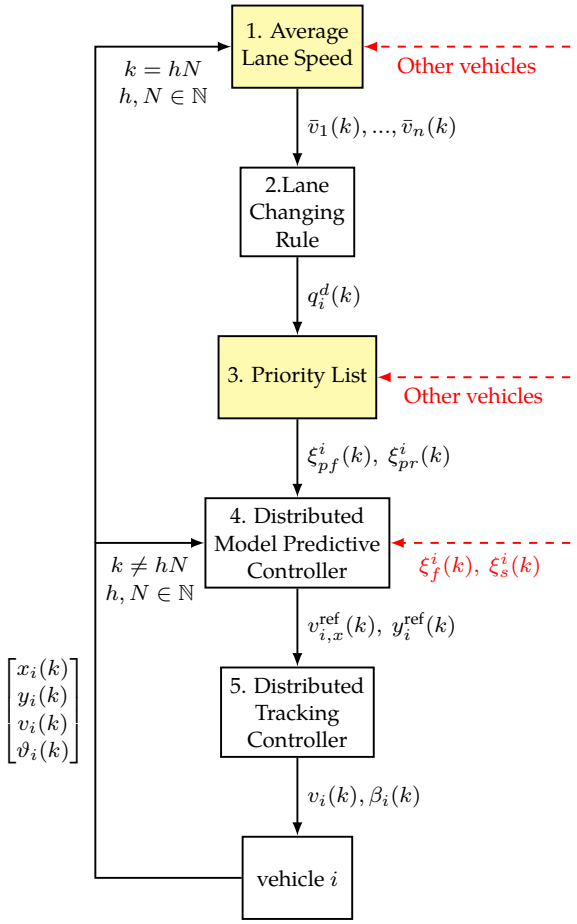


Figure 3.14: Hierarchical Control Structure.

Fig. 3.14 represents the proposed distributed control scheme. By employing such a scheme, agents reach two consecutive agreements: first, on the cruising speed of each lane, then, on when agents can change lanes. That is to say that, initially, all vehicles in one lane agree on a cruising speed for that lane. Knowing this agreed speed, each vehicle can decide whether to change lanes according to its own goals. If a vehicle, indeed, decides to change to another lane, it has to agree

with other traffic participants where and when this change will occur. These two phases occur subsequently every $N \in \mathbb{N}_0$ time steps.

In between iterations hN and $(h+1)N$, $h \in \mathbb{N}_0$, an on-board controller computes an optimal reference trajectory leading to the agreed lane for vehicle i while avoiding collisions. An onboard controller, designed for a discrete time bicycle model, tracks this desired trajectory.

3.2.3 *Average Consensus for Lane Speed*

The first milestone of the proposed controller is to let vehicles in the same lane negotiate a common lane speed. Traffic jam results from string instability; this derives from the fact that any perturbation of leading vehicles' velocity results in amplified fluctuations in velocity (and position) of the following vehicles, see, e.g., [93]. Per contra, if vehicles in a lane agree on a common speed, and this is kept along that lane, traffic jam is avoided and throughput is increased, see, e.g., [49][7]. It is immediate to understand that consensus is a valid tool to the end of an agreement among vehicles.

[7] The main conclusion of this work is that jam dissolution is possible by informing, in real-time, following vehicles about the speed of leading vehicles.

Consider a lane, say q $(1 \leq q \leq n)$. All vehicles in lane q at sampling time k are in set $\mathcal{L}_q(k) := \{i \in \mathcal{N} \mid q_i(k) = q\}$. Each vehicle $i \in \mathcal{L}_q(k)$ aims at agreeing on a common velocity in lane q by distributively negotiating with neighbouring vehicles (e.g., the vehicles directly in front and behind vehicle i). The set of neighboring vehicles is denoted by $\mathcal{N}_i(k)$.

Vehicle i iterates the following consensus protocol:

$$w_i^{\kappa+1} = w_i^{\kappa} + \frac{\sigma}{|\mathcal{N}_i(k)|} \sum_{j \in \mathcal{N}_i(k)} (w_j^{\kappa} - w_i^{\kappa}),\ \kappa = 0, 1, \ldots \tag{3.19}$$

where κ is the iteration index, $\sigma \in (0, 1)$ is a design parameter, and, $\forall i \in \mathcal{L}_q(k)$, w_i is initialized such that $w_i^0 = v_i^d$.

Proposition 13. *Vehicles cruising along lane q (in set $\mathcal{L}_q(k)$) iterate protocol (3.19). If the underlying network topology is strongly connected,*

$$\forall i, j \in \mathcal{L}_q(k),\ \lim_{\kappa \to \infty} w_i^{\kappa} = \lim_{\kappa \to \infty} w_j^{\kappa} = \bar{v}_q(k), \tag{3.20}$$

i.e. consensus is asymptotically achieved and the desired speed for that lane is $\bar{v}_q(k)$.

Proof. By the general results about consensus already presented in Chapter 2, by iterating (3.19) in a strongly connected network, linear average consensus is achieved asymptotically, i.e.,

$$\bar{v}_q(k) = \frac{1}{|\mathcal{L}_q(k)|} \sum_{i \in \mathcal{L}_q(k)} v_i^d.$$

□

Remark 5. *Protocol (3.19) guarantees that average-consensus is asymptotically achieved. However, in practice, the algorithm needs to be stopped after a finite number of iterations. Conditions for stopping a consensus protocol on the basis of local information are described in* [114].

Remark 6. *Additionally, we want to ensure that the agreed speed on lane q satisfies, $\forall k \in \mathbb{N}_0$,*

$$\bar{v}_q(k) \leq \bar{v}_{q+1}(k) \qquad 1 \leq q \leq n-1 \tag{3.21}$$

and

$$\bar{v}_q(k) \leq v_{\text{legal}} \qquad 1 \leq q \leq n. \tag{3.22}$$

This implies that the speeds do not exceed the legal limit v_{legal} and enforces an ordering over the lanes. To achieve this, the consensus values are updated as

$$\bar{v}_n(k) \leftarrow \min(\bar{v}_n(k), v_{\text{legal}}) \tag{3.23}$$

$$\bar{v}_q(k) \leftarrow \min(\bar{v}_q(k), \bar{v}_{q+1}(k)) \qquad 1 \leq q \leq n-1. \tag{3.24}$$

3.2.4 Lane Changing Rule

At this point, each vehicle $i \in \mathcal{L}_q(k)$ knows the negotiated lane speed $\bar{v}_q(k)$. The decision whether to keep the current lane or move into another (faster or slower) can be modeled by a heuristics. In fact, if the difference between the desired speed and the agreed lane speed crosses certain thresholds, vehicles will decide to move into the faster (left) lane or slower (right) lane. This lane changing value is captured in Figure 3.15, where $q_i^d(k)$ is the desired lane of vehicle i.

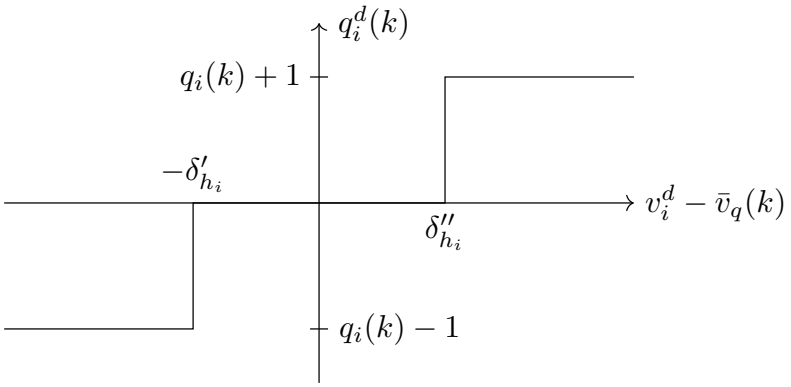


Figure 3.15: Lane Changing Rule.

Given vehicle $i \in \mathcal{L}_q(k)$, if the difference between the desired speed v_i^d and the agreed lane speed $\bar{v}_q(k)$ exceeds $\delta''_{h_i} \in \mathbb{R}_{>0}$, then vehicle i will move into the faster lane, i.e., $q_i^d(k) = q_i(k)+1$, if this exists. If the difference between the desired speed v_i^d and the agreed lane speed $\bar{v}_q(k)$ falls below $-\delta'_{h_i}$, $\delta'_{h_i} \in \mathbb{R}_{>0}$, then vehicle i will move into the slower lane, i.e., $q_i^d(k) = q_i(k)-1$, if this exists. If $-\delta'_{h_i} \leq (v_i^d - \bar{v}_q(k)) \leq \delta''_{h_i}$, vehicle i keeps the current lane.

3.2.5 Priority List

At time step $k \in \mathbb{N}_0$, each vehicle $i \in \mathcal{N}$ has to distributively negotiate its position in the desired lane (both in case $q_i(k) = q_i^d(k)$ and $q_i(k) \neq q_i^d(k)$) with all vehicles aiming at the same lane, i.e., all $j \in \mathcal{N} \setminus \{i\}$,

$q_j^d(k) = q_i^d(k)$. Note that vehicles in the set

$$\{j \in \mathcal{N} \setminus \{i\} \mid q_j(k) = q_i^d(k) \neq q_j^d(k)\} \tag{3.25}$$

are not considered at this point by vehicle $i \in \mathcal{N}$[8]. Let's consider a lane q ($1 \leq q \leq n$) and let's define the set of vehicles aiming at that lane as

$$\mathcal{L}_q^d(k) := \{i \in \mathcal{N} \mid q_i^d(k) = q\}.$$

The objective is to sort vehicles inside of $\mathcal{L}_q^d(k)$ in a distributed fashion based on some criterion. The consensus-based auction algorithm presented in Section 3.1.2 for the automation of road intersections, i.e., CBAA-M[9], is employed also in this context here.

Let each vehicle's bid, i.e., $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $c_i(k)$, be a linear combination of i's longitudinal position and cruising speed, i.e.,

$$c_i(k) = x_i(k) + \gamma v_i(k), \tag{3.26}$$

where $\gamma \in \mathbb{R}_{>0}$. By Proposition 12 and under the assumption that the underlying network topology is connected, in a finite amount of iterations, each vehicle i acquires $\mathbf{v}^*(k)$, which is a list of agents sorted according to their bids. Note that $\mathbf{v}^*(k)$ sorts agents in $\mathcal{L}_q^d(k)$ according to their bids.

[8] These vehicles will be considered by the MPC for collision avoidance purposes as *side vehicles*.

[9] Designed and named by the candidate earlier in Molinari and Raisch [59].

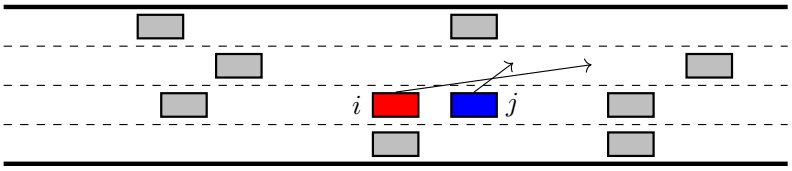


Figure 3.16: A possible unreasonable trajectory that might result from the distributed auction.

At first glance, a coherency problem might be expected. Consider two vehicles, say i and j, with

$$q_i(k) = q_j(k),\ q_i^d(k) = q_j^d(k),\ \text{and}\ x_j(k) > x_i(k),$$

(thus j being in front of i). In case $c_i(k) > c_j(k)$, vehicle i would end up in front of j on the new lane. The trajectory to achieve this goal is unreasonable, as indicated in Figure 3.16. However, by the following proposition, this problem cannot occur if γ is sufficiently large.

Proposition 14. *If two vehicles i and j exist, so that*

$$q_i(k) = q_j(k),\ q_i^d(k) = q_j^d(k),\ \text{and}\ x_j(k) > x_i(k),$$

then $c_j(k) > c_i(k)$ if

$$\gamma > \frac{(v_i(k) - v_j(k))}{\tilde{D}}. \tag{3.27}$$

This is an interpretation of Proposition 14. Say $q_i(k) = q$. Since j is frontal, and trivially assuming that no collision is happening, $x_j(k) > x_i(k) + \tilde{D}$. $c_j(k) > c_i(k)$ holds if and only if

$$\gamma > \frac{(v_i(k) - v_j(k))}{\tilde{D}}.$$

This is a straight consequence of (3.26). As vehicles i and j are on the same lane, $v_i(k) \simeq \bar{v}_q(k) \simeq v_j(k)$, i.e., the difference $v_i(k) - v_j(k)$ is small in magnitude. Hence, the condition in (3.27) is not restrictive.

3.2.6 *Distributed Model Predictive Controller*

Each vehicle $i \in \mathcal{N}$ at every time step $k \in \mathbb{N}_0$ retrieves the following pieces on information about the surrounding traffic. Its frontal vehicle in the current lane is $\xi_f^i(k) \in \mathcal{N}$. Moreover, vehicle i has target lane $q_i^d(k)$ (updated every N discrete-time steps) and future frontal and rear vehicles (as in list $\mathbf{v}^*(k)$) are, respectively, $\xi_{pf}^i(k) \in \mathcal{N}$ and $\xi_{pr}^i(k) \in \mathcal{N}$. Another vehicle needs to be taken into account to the extent of avoiding collisions: let, in fact, $\xi_s^i(k)$ be a vehicle cruising next to ego-vehicle i along lane $q_i^d(k)$. If this vehicle aims for another lane, it will not be considered in the priority list (see (3.25)), but still needs to be taken into account for avoiding collisions. Figure 3.17 helps to visualize all considered vehicles. Since vehicle i is only interested in lanes $q_i(k)$ and $q_i^d(k)$, avoiding collisions with these four vehicles is sufficient. Information from these vehicles is obtained by V2V communication.

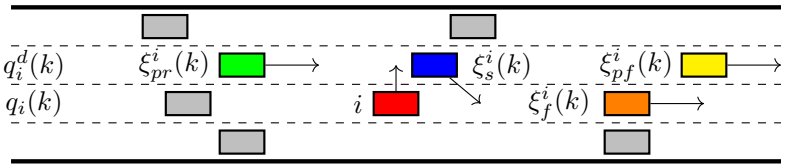


Figure 3.17: Vehicles considered in the optimal control problem formulation for vehicle i at time $k \in \mathbb{N}_0$. Arrows display their desired direction.

$x_{k+h|k}^{j,i}$ denotes the prediction (done by vehicle i) of variable x_j (i.e., the longitudinal position of vehicle j) for time step $k+h$ given information up to time k. An analogous notation is used for all other predicted state and input variables. Variable $q_{k+h|k}^{i,i}$, $h \in \{0 \dots H\}$, is the predicted lane of vehicle i at step $k+h$ given information up to time k. The optimal control problem is formulated for the following simple prediction model, in which longitudinal and lateral dynamics are decoupled and speed is the control variable:

- For vehicle i, $\forall p \in \{0 \dots H\}$,

$$x_{k+p+1|k}^{i,i} = x_{k+p|k}^{i,i} + T_s v_{k+p|k}^{i,i} \quad , \tag{3.28}$$

$$q_{k+p+1|k}^{i,i} \in \{q_i(k), q_i^d(k)\} \quad , \tag{3.29}$$

where, $\forall p \in \{0 \dots H\}$, $v_{k+p|k}^{i,i}$ is i's predicted longitudinal speed (first decision variable) computed at time k. $q_{k+p|k}^{i,i}$, $p \in \{1 \dots H+1\}$ (second decision variable) is a positive integer representing the predicted lane. One possible way[10] to tackle nonconvexity is, in fact, to employ integer variables, see, e.g., [62]. However, involving integer variables in the optimal control problem leads to mixed integer programming, which is notorious for high computation complexity. Techniques for reducing the number of integer variables thus facilitating real-time implementability, see, e.g., [62], can be employed. Predictions are initial-

[10] Alternative to the approach of Section 3.1.

$x^{j,i}_{k+h\|k}$	prediction of x_j done by i for time $k+h$ given k
$v^{j,i}_{k+h\|k}$	prediction of v_j done by i for time $k+h$ given k
$q^{j,i}_{k+h\|k}$	j's predicted lane for time $k+h$ given k according to i
$q_i(k)$	i's current lane at time k
$q_i^d(k)$	i's desired lane at time k
$\tilde{D}$	Safety distance
$\underline{v}$, $\overline{v}$	Minimum and maximum speed
$\underline{a}$, $\overline{a}$	Minimum and maximum acceleration
$\bar{v}_{q_i(k)}(k)$	Agreed speed in the current lane

Table 3.1: Symbols used in the MPC formulation.

ized as follows:

$$x^{i,i}_{k+0|k} = x_i(k) \quad , \tag{3.30}$$

$$q^{i,i}_{k+0|k} = q_i(k) \quad , \tag{3.31}$$

$$v^{i,i}_{k+0|k} = v_i(k) \quad . \tag{3.32}$$

Each vehicle $i \in \mathcal{N}$ broadcasts the sequence of its predicted longitudinal positions and lanes, i.e. $x^{i,i}_{k+p|k}$ and $q^{i,i}_{k+p|k}$, $p \in \{1 \dots H+1\}$, to other vehicles. Table 3.1 summarizes all variables and parameters employed in this section.

■ Therefore, vehicle i knows the predicted longitudinal positions and lane numbers for each vehicle $j \in \{\xi^i_f(k), \xi^i_s(k), \xi^i_{pr}(k), \xi^i_{pf}(k)\}$, i.e., $\forall p \in \{0 \dots H\}$,

$$x^{j,i}_{k+p|k} = x^{j,j}_{k+p|k-1} \quad , \tag{3.33}$$

$$q^{j,i}_{k+p|k} = q^{j,j}_{k+p|k-1} \quad . \tag{3.34}$$

Proposition 15. *The following set of constraints aim at avoiding collisions over the prediction horizon:*
$\forall p \in \{0 \dots H\}, \forall j \in \mathcal{N} \setminus \{i\}$,

$$|x^{i,i}_{k+p|k} - x^{j,i}_{k+p|k}| \geq \tilde{D} \quad \vee \quad q^{i,i}_{k+p|k} \neq q^{j,i}_{k+p|k} \tag{3.35}$$

Proof. The proof follows immediately from (3.18). □

Constraints (3.35) are non-convex and non-linear and will be reformulated to be incorporated into the optimal control problem. As stated above, one possible approach is to employ integer variables. In fact, the "logic OR" statement is traditionally reformulated by using one or more binary variables and the so-called *Big-M reformulation*, see [112]. Accordingly, in what follows, let $M \in \mathbb{R}_{>0}$ be a large positive number. Further details are in Proposition 16. Furthermore, we decompose the safety requirement (3.35) by distinguishing between three possible scenarios that vehicle i can encounter, namely left-turning, right-turning, and going straight. The following variable is employed to formally distinguish between the scenarios:

$$\epsilon := q_i^d(k) - q_i(k) = \begin{cases} 1 & \text{for left-turning} \\ -1 & \text{for right-turning} \\ 0 & \text{for going straight} \end{cases} \tag{3.36}$$

Left-turning

As long as the frontal vehicle $\xi_f^i(k)$ is predicted to be in lane $q_i(k)$ and vehicle i has not implemented the desired lane change, a minimum distance of $\tilde{D}$ between both vehicles is required. This requirement can be relaxed when vehicle i has changed lanes. Formally:

$$\forall p \in \left\{ p \in \{0 \dots H\} \mid q_{k+p|k}^{\xi_f^i(k),i} = q_i(k) \right\},$$
$$x_{k+p|k}^{\xi_f^i(k),i} - x_{k+p|k}^{i,i} \geq \tilde{D} - \epsilon M \left(q_{k+p|k}^{i,i} - q_i(k) \right). \quad (3.37)$$

When changing lane, vehicle i should have sufficient maneuver space. This translates into requiring a minimal distance towards its future frontal and rear vehicles, i.e., $\xi_{pf}^i(k)$ and $\xi_{pr}^i(k)$. This requirement must hold for all time instants where these vehicles are on vehicles i's desired lane. Furthermore, the requirement can be relaxed if vehicle i has not yet implemented the desired lane switch. Formally,

$$\forall p \in \left\{ p \in \{0 \dots H\} \mid q_{k+p|k}^{\xi_{pf}^i(k),i} = q_i^d(k) \right\},$$
$$x_{k+p|k}^{\xi_{pf}^i(k),i} - x_{k+p|k}^{i,i} \geq \tilde{D} - \epsilon M \left(q_i^d(k) - q_{k+p|k}^{i,i} \right) \quad (3.38)$$

and

$$\forall p \in \left\{ p \in \{0 \dots H\} \mid q_{k+p|k}^{\xi_{pr}^i(k),i} = q_i^d(k) \right\},$$
$$x_{k+p|k}^{i,i} - x_{k+p|k}^{\xi_{pr}^i(k),i} \geq \tilde{D} - \epsilon M \left(q_i^d(k) - q_{k+p|k}^{i,i} \right). \quad (3.39)$$

To also rule out collisions with cars on the lane to the left driving side-by-side with vehicle i, we enforce the following constraints:

$$\forall p \in \left\{ p \in \{0 \dots H\} \mid q_{k+p|k}^{\xi_s^i(k),i} = q_i(k) \right\},$$
$$x_{k+p|k}^{\xi_s^i(k),i} - x_{k+p|k}^{i,i} \geq \tilde{D} - \epsilon M \left(q_{k+p|k}^{i,i} - q_i(k) \right) \quad (3.40)$$

and

$$\forall p \in \left\{ p \in \{0 \dots H\} \mid q_{k+p|k}^{\xi_s^i(k),i} = q_i^d(k) \right\},$$
$$x_{k+p|k}^{\xi_s^i(k),i} - x_{k+p|k}^{i,i} \geq \tilde{D} - \epsilon M \left(q_i^d(k) - q_{k+p|k}^{i,i} \right). \quad (3.41)$$

(3.40) implies that if $\xi_s^i(k)$ cuts into lane $q_i(k)$, it does so in front of vehicle i and a minimum distance $\tilde{D}$ is enforced between this vehicle and vehicle i. This requirement is relaxed if vehicle i implements the desired lane change. (3.41) implies that if vehicle i implements its lane change, with $\xi_s^i(k)$ still on vehicles i's target lane, vehicle i cuts in behind $\xi_s^i(k)$ and, again, a minimum distance $\tilde{D}$ is enforced.

Right-turning

Collision avoidance constraints for this scenario are analogous to constraints (3.37)-(3.41), but with $\epsilon = -1$ and the relevant side vehicle traveling on the lane to the right of vehicle i.

Go straight ($q_i^d(k) = q_i(k)$)

In this scenario, only $\xi_f^i(k)$ and $\xi_{pf}^i(k)$ need to be considered. Hence, collision avoidance requirements correspond to (3.37)-(3.38) with $\epsilon = 0$.

Proposition 16. *Constraints* (3.37)-(3.41) *imply* (3.35).

Proof. The proof is straightforward. □

In addition to the safety constraints (3.37)-(3.41), speed and acceleration of vehicle i can be constrained as follows (with $\bar{v},\ \underline{v},\ \bar{a} \in \mathbb{R}_{>0}$ and $\underline{a} \in \mathbb{R}_{<0}$):

$$\forall p \in \{0 \dots H\}, \quad v_{k+p|k}^{i,i} \in [\underline{v}, \overline{v}] \quad , \tag{3.42}$$

$$\forall p \in \{1 \dots H\}, \quad T_s \underline{a} \leq v_{k+p|k}^{i,i} - v_{k+p-1|k}^{i,i} \leq T_s \bar{a} \quad . \tag{3.43}$$

Keeping a larger safety distance is beneficial for safety reasons. Vehicle i can be encouraged to keep a larger distance by rewriting $\tilde{D}$ as a variable of the optimal control problem, so that it incorporates a slack variable; this is a standard approach for collision avoidance constraints, see, e.g., [42]. Formally,

$$\tilde{D} \geq \tilde{D}_0 + \delta, \tag{3.44}$$

$$\delta \geq 0, \tag{3.45}$$

where $\tilde{D}_0 \in \mathbb{R}_{>0}$ is a minimum safety distance that vehicles must hold. In the cost function, δ will be negatively weighted, such that keeping a larger δ will be encouraged.

Accordingly, the cost function is formulated as

$$\begin{aligned} J = \sum_{p=1}^{H} & a_1 \left(v_{k+p|k}^{i,i} - v_{k+p-1|k}^{i,i} \right)^2 \\ & + a_2 \left(v_{k+p|k}^{i,i} - \bar{v}_{q_i(k)}(k) \right)^2 \\ & + a_3 \left(q_{k+p|k}^{i,i} - q_i^d(k) \right)^2 - a_4 \delta, \end{aligned} \tag{3.46}$$

where $a_1,\ a_2,\ a_3,\ a_4 \in \mathbb{R}_{>0}$ are design parameters. The first term of the sum discourages high values of acceleration and deceleration, thus contributing to onboard comfort. The second term punishes deviations from the agreed lane speed. The third term encourages vehicle i to move into the desired lane as soon as the constraints allow for it. Finally, the last term rewards large δ, hence contributing to increase safety. The optimal control problem becomes:

$$\min_{\substack{v_{k+1|k}^{i,i} \cdots v_{k+H|k}^{i,i} \\ q_{k+1|k}^{i,i} \cdots q_{k+H+1|k}^{i,i}}} \quad J$$

s.t.	prediction model	(3.28) – (3.29)
	initial conditions	(3.30) – (3.32)
	collision avoidance	(3.37) – (3.41)
	safety distance formulation	(3.44) – (3.45)
	input and state constraints	(3.42) – (3.43)

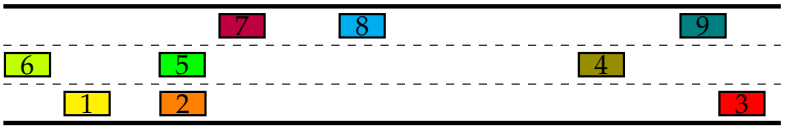


Figure 3.18: Initial position of vehicles considered in the simulation.

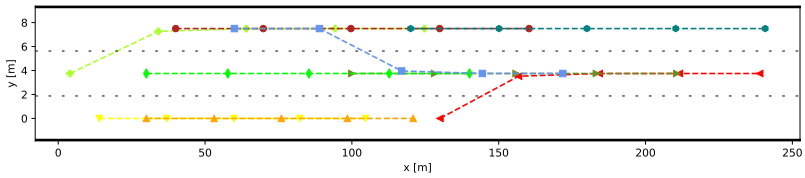


Figure 3.19: Positions (sampled each 20 [ms]) of vehicles in the highway.

According to the MPC philosophy, only the initial portions $v^{i,i}_{k+1|k}$ and $q^{i,i}_{k+1|k}$ of the optimal solution will be used as a reference for the distributed tracking controllers.

An MPC controller is recursively feasible if, for all feasible initial states and for all optimal sequences of control inputs, the MPC optimization problem remains feasible during the whole horizon (see [54]).

3.2.7 *Distributed Tracking Controller*

In the following, let $v^{\text{ref}}_{i,x}(k) := v^{i,i}_{k+1|k}$ be the reference longitudinal speed and

$$y^{\text{ref}}_i(k) := q^{i,i}_{k+1|k} w_h$$

the reference lateral coordinate obtained from the MPC. Longitudinal and lateral control can be decoupled. In fact, under the assumption of a small yaw angle $\vartheta_i(k)$ (realistic for a highway scenario), the longitudinal dynamics becomes

$$\begin{aligned} x_i(k+1) &= x_i(k) + T_s v_i(k) \cos(\vartheta_i(k)) \\ &\simeq x_i(k) + T_s v_i(k). \end{aligned} \tag{3.47}$$

By this, the optimal longitudinal speed $v^{\text{ref}}_{i,x}(k)$ (see (3.28)) can serve as the cruising speed $v_i(k)$, i.e. $v_i(k) = v^{\text{ref}}_{i,x}(k)$.

Regarding the lateral dynamics, a controller computes the steering angle $\beta_i(k)$, such that $y^{\text{ref}}_i(k)$ is tracked. By (3.16), the discrete-time version of the lateral dynamics, where $v_i(k) = v^{\text{ref}}_{i,x}(k)$, is

$$\begin{cases} y_i(k+1) = y_i(k) + T_s v^{\text{ref}}_{i,x}(k) \sin(\theta_i(k)) \\ \theta_i(k+1) = \theta_i(k) + \frac{v^{\text{ref}}_{i,x}(k) T_s}{L} \tan(\beta_i(k)) \end{cases} \tag{3.48}$$

For small angles, $\sin\theta \simeq \theta$, then

$$\begin{cases} y_i(k+1) = y_i(k) + T_s v^{\text{ref}}_{i,x}(k) \theta_i(k) \\ \theta_i(k+1) = \theta_i(k) + \frac{v^{\text{ref}}_{i,x}(k) T_s}{L} \beta_i(k) \end{cases}, \tag{3.49}$$

where the variables now denote variations from the stationary point. A proportional controller K for reference tracking is calculated by *pole placement* from (3.49), see [5, pg. 196].

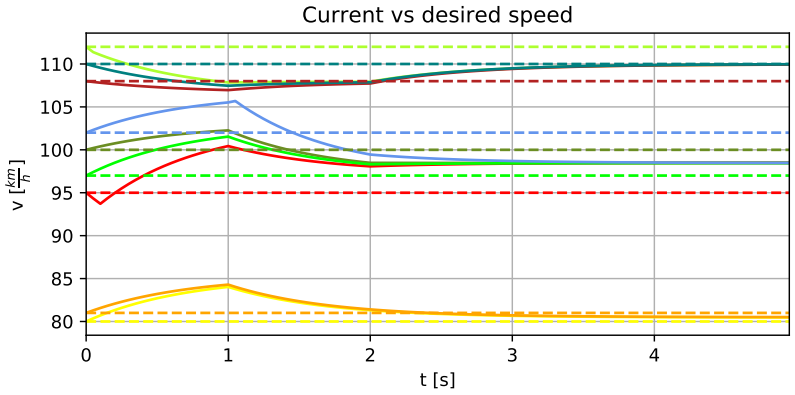


Figure 3.20: Desired speeds (dashed lines) and current speeds (solid lines) of vehicles throughout the simulation.

Figure 3.21: Rear-end vehicle's distance to the closest vehicle.

3.2.8 *Simulations*

The scenario represented in Figure 3.18, with $m = 9$ vehicles cruising along a highway composed of $n = 3$ lanes, is simulated. Desired speeds are in Table 3.2. Each vehicle has an initial speed equal to the desired, i.e. $\forall i \in \mathcal{N},\ v_i^d = v_i(0)$. Parameters considered in simulation are in Table 3.3. The simulation is run for 5 [s] (which corresponds to 100 intervals). Figure 3.19 shows vehicles' positions throughout the simulation. Vehicle 6 has the highest desired speed, much higher than vehicles 4 and 5 (starting on the same lane). It moves into the third (left) lane immediately. As it moves, vehicle 8 decides to move into a slower lane (the middle lane). In fact, as one can see in Figure 3.20, up to $t = 1$ [s], vehicle 8 increases its speed to remain on the third lane (where the agreed speed equals $\frac{108+102+110}{3} = 106.7\ [\frac{\text{km}}{\text{h}}]$). However, as vehicle 6 moves into that lane (at $t \simeq 1$ [s]), it affects the average speed negotiation (new agreed speed equal $\frac{108+102+110+112}{4} = 108\ [\frac{\text{km}}{\text{h}}]$), so that vehicle 8's decision rule makes it move into the second lane (from Table 3.3, $\delta'_{h_i} = \delta''_{h_i} = 5\ [\frac{\text{km}}{\text{h}}]$). At the next negotiation time (since $N = 20$, it is at $t = 2$ [s]), the agreed speed for the third lane is then $\frac{108+112+110}{3} = 110\ [\frac{\text{km}}{\text{h}}]$. This can be identified in Figure 3.20, since vehicles in the third lane accelerate to $110\ [\frac{\text{km}}{\text{h}}]$ after $t = 2$ [s].

$v_1^d = 80.0\ [\frac{\text{km}}{\text{h}}]$, $v_2^d = 81.0\ [\frac{\text{km}}{\text{h}}]$, $v_3^d = 95.0\ [\frac{\text{km}}{\text{h}}]$
$v_4^d = 100.0\ [\frac{\text{km}}{\text{h}}]$, $v_5^d = 97.0\ [\frac{\text{km}}{\text{h}}]$, $v_6^d = 112.0\ [\frac{\text{km}}{\text{h}}]$
$v_7^d = 108.0\ [\frac{\text{km}}{\text{h}}]$, $v_8^d = 102.0\ [\frac{\text{km}}{\text{h}}]$, $v_9^d = 110.0\ [\frac{\text{km}}{\text{h}}]$

Table 3.2: Vehicles' desired speeds.

Table 3.3: Parameters.

$\delta'_{h_i} = \delta''_{h_i} = 5$ [$\frac{\text{km}}{\text{h}}$], $w_h = 375$ [m], $L = 3$ [m], $\tilde{D}_0 = 8$ [m]
$a_1 = 100$, $a_2 = 1$, $a_3 = 0.1$, $a_4 = 0.1$,
$H = 10$, $T_s = 0.05$ [s], $N = 20$,
$\gamma = 0.2$

Table 3.4: Initial longitudinal positions. Initial lateral position can be obtained from Figure 3.18.

$x_1(0) = 14.0$ [m], $x_2(0) = 30.0$ [m], $x_3(0) = 130.0$ [m],
$x_4(0) = 100.0$ [m], $x_5(0) = 30.0$ [m], $x_6(0) = 4.0$ [m],
$x_7(0) = 40.0$ [m], $x_8(0) = 60.0$ [m], $x_9(0) = 120.0$ [m]

As vehicle 8 decides to move into the second lane, it negotiates an entry point in between vehicles 5 and 4 (according to the bidding parameter γ in Table 3.3). From Figure 3.21, where minimum frontal distances of each vehicle are plotted, one can see the impact of vehicle 8 lane change after $t = 1$ [s]. In fact, at $t \simeq 1.25$ [s], minimum distance of vehicle 5 towards other vehicles drops down to 31 [m] (since vehicle 8 cuts in in front of it). Also the new minimum distance of vehicle 8 drops down to 39 [m], since it cuts in behind vehicle 4. After this moment, since all vehicles on the same lane are driving with the same agreed speed, rear-end minimum distances hold constant. Collisions are clearly avoided, since inter-vehicular distances are always larger than 15 [m].

4
Exploiting Wireless Interference

This Chapter contains background information about the employed communication technology at its state of the art. It is based on extensive discussions with Navneet Agrawal, MSc.

As introduced in Section 2, this thesis focuses on designing consensus protocols bringing a multi-agent system to an agreement, while exploiting the superposition property of the wireless channel for communication. The goal of this chapter is giving a brief overview of all involved communication-related concepts, which are, as explained above, not the main focus of this thesis. A detailed contribution about the communication technology can be found in [12, 45]. The complete channel model obtained in Molinari et al. [68] integrates what presented in this section.

4.1 Orthogonal Channel Access Methods in a nutshell

In recent years, an increasing number of publications, see, e.g., [34], have tried to quantify the wireless communication effort required by the general consensus protocol (2.13). A consensus protocol is more efficient if it can achieve the same result by using less resources. For a

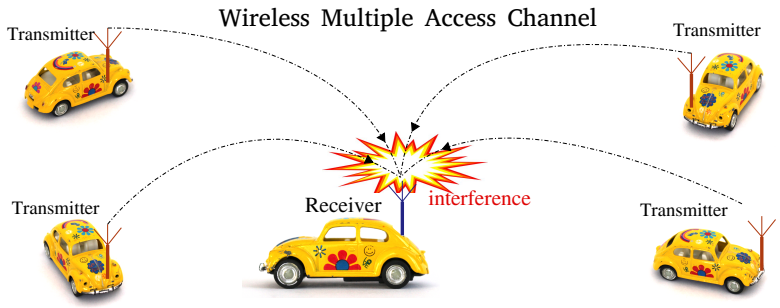


Figure 4.1: Interference is the physical phenomenon obtained when multiple users access synchronously the same channel frequency. Credits to Navneet Agrawal for the image.

standard implementation of (2.13), each agent $i \in \mathcal{N}$ needs to obtain, at every iteration $k \in \mathbb{N}_0$, all elements in set $\{x_j(k)\}_{j \in \mathcal{N}_i(k)}$. As in Figure 4.1, if multiple transmitters simultaneously access the same shared medium in the same frequency, their signals will interfere. In fact, by [107, pg. 100], the wireless channel is a shared broadcast medium;

letting multiple users access the same channel frequency spectrum simultaneously results in interference. Physically, the electromagnetic waves broadcast by a set of transmitters in the same frequency band superimpose (sum) at the receiver.

This phenomenon has been traditionally combatted, indeed allowing receivers to obtain uncorrupted signals. Each signal arriving at the destination without interfering with other signals is said to be *orthogonally transmitted*. Therefore, under a communication technology point of view, at every iteration $k \in \mathbb{N}_0$, the standard implementation of (2.13) requires

$$\sum_{i=1}^{n} |\mathcal{N}_i(k)| \tag{4.1}$$

orthogonal transmissions, that is to say, every link-to-link transmission between any pair of agents $i - j$ needs to be free of interference with all other individual transmissions of data. This is the case, for example, if agents access the channel at different times (TDMA, Time Division Multiple Access), or at different frequencies (FDMA, Frequency Division Multiple Access). In what follows, such multiplexing techniques will be referred to as *orthogonal channel access methods*. Figure 4.2 illustrates the idea behind orthogonal channel access methods. However, enforcing orthogonal transmissions comes with a cost in terms of employed wireless resources (with the number of orthogonal transmissions being given by (4.1)), which are traditionally not taken into account in most consensus literature.

4.2 Wireless Multiple Access Channel Model

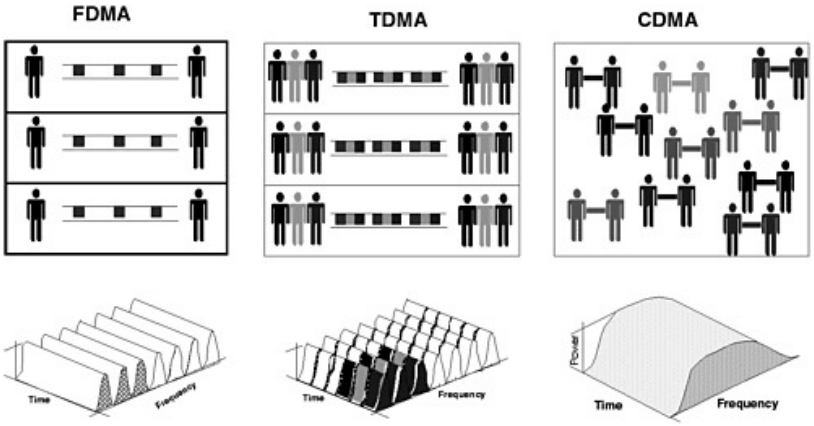


Figure 4.2: Orthogonal channel access methods.

According to the intuition in [35], it is possible to exploit interference to get to consensus. This is achieved by designing a wireless communication system that exploits interference. Exploiting interference for achieving consensus leads to saving wireless resources: the orthogonal transmissions quantified in (4.1) are no longer needed. The way the received information, corrupted by interference can still be used for achieving consensus is the topic of this work.

For describing the input-output relationship in the shared wireless channel where two or more users simultaneously transmit over the same frequency, we introduce the so-called *Wireless Multiple Access Channel* (WMAC) model.

Definition 14 (WMAC)**.** *Let $\mathcal{N}_i \subset \mathcal{N}$ be a subset of agents transmitting to a designated agent $i \in \mathcal{N}$ at iteration $k \in \mathbb{N}_0$. Each agent $j \in \mathcal{N}_i$ at iteration $k \in \mathbb{N}_0$ transmits $\omega_j(k) \in \mathbb{R}$. The signal obtained by the receiver is $z_i(k) \in \mathbb{R}$, computed as*

$$z_i(k) := \sum_{j \in \mathcal{T}} \xi_{ij}(k)\, \omega_j(k) + \eta_i(k), \tag{4.2}$$

where, $\forall k \in \mathbb{N}_0$, $\forall j \in \mathcal{N}_i$, $\xi_{ij}(k) \in \mathbb{R}$ is the fading coefficient that captures the fading effect between the transmitter j and the receiver i and $\eta_i(k)$ the receiver noise.

One can see that, in the presented model, there are three phenomena that impact on the correct reception of the desired quantity: fading (multiplicative noise), superposition, and additive noise. Notwithstanding these three nonidealities, in the remainder of this work, we will present methods to achieve consensus by exploiting interference. Throughout recent years, various simplifying assumptions on the WMAC model have been hold.

4.2.1 Ideal Channel

The early works in the field considered the fading channel to be *ideal*, i.e., fading coefficients to be 1 and additive noise to be null. This way, the WMAC yields the real-valued signal

$$z_i(k) = \sum_{j \in \mathcal{T}} \omega_j(k). \tag{4.3}$$

Only the superposition effect (and not fading nor noise corruption) was considered. This was the case for a collection of papers concerning average consensus, e.g., [35, 107]. Also Molinari et al. [64] used this model.

4.2.2 Noiseless Real-valued Fading Channel

A more realistic model considers the effect of a real-valued fading channel. Starting from the WMAC in Definition 14, it is possible to simplify it so that the obtained real-valued signal $z_i(k)$ is

$$z_i(k) = \sum_{j \in \mathcal{T}} \xi_{ij}(k)\omega_j(k), \tag{4.4}$$

with $\forall i, j \in \mathcal{N}$, $\xi_{ij}(k) \in \mathbb{R}_{>0}$ by designing a communication system (and holding some assumptions on the communication technology) as in [12]. In this model, the two considered effects linked to wireless transmission are fading and superposition.

The first papers in literature using this model for consensus are Molinari et al. [63] for the average consensus and Molinari et al. [68]

for the max-consensus. After these, other researchers have contributed to the field, see, e.g., [53, 115].

Assumption 4. *In this work, we assume the real-fading coefficients to be independent realizations (across time and agents) of the same uniform distribution, i.e.,* $\forall k \in \mathbb{N}_0$, $\forall j \in \mathcal{N}_k$, $\xi_{ij}(k) \sim \mathcal{U}(0, a]$, *where* $a \in \mathbb{R}_{>0}$.

5
Average Consensus Over the Wireless Channel

Let $\mathcal{N} = \{1 \dots n\}$ be the set of intercommunicating agents, labeled 1 through $n \in \mathbb{N}$. They are seeking for an agreement over a variable of common interest (see Section 2.3.1 for an introductory analysis of consensus problems). To this end, they synchronously exchange information at discrete-time steps $k \in \mathbb{N}_0$ over the wireless channel (see Section 4 for an introduction to the topic). Let agents update the information state $x_i(k)$ according to the following equation:

$$\forall i \in \mathcal{N}, \qquad x_i(k+1) = x_i(k) + u_i(k), \tag{5.1}$$

where $x_i : \mathbb{N}_0 \mapsto \mathbb{R}$ is the agent's state and $u_i : \mathbb{N}_0 \mapsto \mathbb{R}$ its input. The goal is designing a combined communication and control system that exploits the superposition property of the wireless channel and let agents reach an agreement in the sense of (2.7).

5.1 Average Consensus Protocol

Partial results of the presented section have been published in Molinari et al. [63].

This section presents a consensus protocol based on the WMAC model (4.4).

5.1.1 Communication Protocol

Each agent $j \in \mathcal{N}$ broadcasts to the channel two orthogonal signals at every iteration $k \in \mathbb{N}_0$, i.e.,

$$\forall k \in \mathbb{N}_0,\ \tau_j(k) := x_j(k) \tag{5.2}$$

and

$$\forall k \in \mathbb{N}_0,\ \tau'_j(k) := 1. \tag{5.3}$$

This is done by employing a MAC of order 2, see, e.g., [36]. According to this WMAC model (see Section 4 for a thorough explanation), each agent $i \in \mathcal{N}$ receives two orthogonal real-valued signals, coming from the superposition of what neighboring agents have transmitted, i.e.,

$$\forall k \in \mathbb{N}_0,\ \upsilon_i(k) := \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)\tau_j(k) = \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)x_j(k) \tag{5.4}$$

and

$$\forall k \in \mathbb{N}_0,\ \upsilon'_i(k) := \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)\tau'_j(k) = \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k). \tag{5.5}$$

Remark 7. *Note that, in case $\mathcal{N}_i(k) = \emptyset$ for some $i \in \mathcal{N}$ and $k \in \mathbb{N}_0$, then we assume both $\upsilon_i(k) = 0$ and $\upsilon'_i(k) = 0$.*

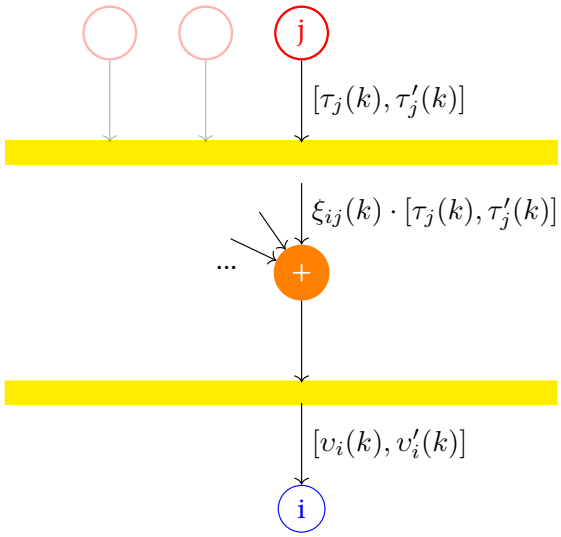


Figure 5.1: Sketch of the designed communication protocol. Agents in set $\mathcal{N}_i(k)$ broadcast to the wireless medium. Agent i receives two orthogonal interfered signals.

5.1.2 Consensus Protocol

The control input to (5.1) at every iteration $k \in \mathbb{N}_0$ is obtained by combining the two received signals, $\upsilon_i(k)$ and $\upsilon'_i(k)$, as follows:

■ $\mathcal{N}_i(k) \neq \emptyset$:

$$\begin{aligned} u_i(k) &= \sigma_i(k) \left(\frac{v_i(k)}{v_i'(k)} - x_i(k) \right) \\ &= \sigma_i(k) \left(\frac{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) x_j(k)}{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)} - x_i(k) \right) \\ &= \sigma_i(k) \left(\sum_{j=1}^{n} h_{ij}(k) x_j(k) - x_i(k) \right), \end{aligned} \tag{5.6}$$

■ $\mathcal{N}_i(k) = \emptyset$:

$$u_i(k) = 0, \tag{5.7}$$

where, $\forall k \in \mathbb{N}_0$, $\forall i \in \mathcal{N}$, $\sigma_i(k) \in [0, 1)$ is the so-called agent's stubbornness parameter and, $\forall i, j \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $h_{ij}(k)$ is the normalized channel coefficient (nonnegative by construction), which is defined as follows:

$$h_{ij}(k) := \begin{cases} \dfrac{\xi_{ij}(k)}{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)} & \text{if } j \in \mathcal{N}_i(k) \\ 0 & \text{otherwise} \end{cases}. \tag{5.8}$$

Observation 1. *By construction and if each agent always has at least one neighbor,*[1]*, i.e.,* $|\mathcal{N}_i(k)| \geq 1$,

[1] In the sense of (2.2).

$$\forall k \in \mathbb{N}_0,\ \forall i \in \mathcal{N}, \quad \sum_{j=1}^{n} h_{ij}(k) = 1. \tag{5.9}$$

By putting (5.6)-(5.7) into (5.1), the information state of agent $i \in \mathcal{N}$, in case $\mathcal{N}_i(k) \neq \emptyset$, evolves according to

$$x_i(k+1) = (1 - \sigma_i(k)) x_i(k) + \sigma_i(k) \sum_{j=1}^{n} h_{ij}(k) x_j(k), \tag{5.10}$$

where, if $\mathcal{N}_i(k) = \emptyset$, $\sigma_i(k)$ needs to be forced to 0, namely,

$$\forall i \in \mathcal{N},\ \forall k \in \mathbb{N}_0,\ |\mathcal{N}_i(k)| = 0 \iff \sigma_i(k) = 0. \tag{5.11}$$

This dynamics discloses the concept of stubbornness. In fact, the closer $\sigma_i(k)$ is to 1, the more agent i relies on the received signal for computing $x_i(k+1)$. Stubbornness $\sigma_i(k)$ close to 0 keeps $x_i(k+1)$ close to the current information state $x_i(k)$.

The term $\sum_{j=1}^{n} h_{ij}(k) x_j(k)$ contains the information states of neighbors. However, being the normalized fading channel coefficients unknown, agents cannot reconstruct the individual information states of their neighbors one by one.

Proposition 17. *If agent $i \in \mathcal{N}$ has, at iteration $k \in \mathbb{N}_0$, at least one neighbor, i.e., $|\mathcal{N}_i(k)| > 0$, then,*

$$\sum_{j=1}^{n} h_{ij}(k) x_j(k) \in \mathfrak{C}\left(\{x_j(k)\}_{j \in \mathcal{N}} \right). \tag{5.12}$$

Proof. The proof is obtained as a consequence of Observation 1. By this, normalized channel coefficients, i.e., $\forall k \in \mathbb{N}_0$, $\{h_{ij}(k)\}_{i,j\in\mathcal{N}}$, can be seen as coefficients of a convex combination. This way, $\sum_{j=1}^{n} h_{ij}(k)x_j(k)$ is a convex combination of $\{x_j(k)\}_{j\in\mathcal{N}}$, which is nonempty by hypothesis. This yields (5.12), thus concluding the proof. □

By this latter result, we obtain that the information state of each agent $i \in \mathcal{N}$ at iteration $k \in \mathbb{N}_0$ is updated to be a convex combination[2] of its current value and a value in the convex hull of neighbors' information states. In the upcoming analysis, we will also design $\sigma_i(k)$, so that it can drive the system towards desired performance.

[2] With convex combination coefficients $(1-\sigma_i(k))$ and $\sigma_i(k)$.

5.1.3 Analysis

The system can be written in compact form as,

$$\forall k \in \mathbb{N}_0,\ \boldsymbol{x}(k+1) = D_n^{\sigma}(k)\boldsymbol{x}(k), \tag{5.13}$$

where, $\forall k \in \mathbb{N}_0$, $\boldsymbol{x}(k) = [x_1(k) \dots x_n(k)]'$ and

$$D_n^{\sigma}(k) := \begin{bmatrix} 1-\sigma_1(k) & \sigma_1(k)h_{12}(k) & \dots & \sigma_1(k)h_{1n}(k) \\ \sigma_2(k)h_{21}(k) & 1-\sigma_2(k) & \dots & \sigma_2(k)h_{2n}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_n(k)h_{n1}(k) & \sigma_n(k)h_{n2}(k) & \dots & 1-\sigma_n(k) \end{bmatrix}. \tag{5.14}$$

Proposition 18. *$\forall k \in \mathbb{N}_0$, $D_n^{\sigma}(k)$ is nonnegative and row-stochastic.*

Proof. Nonnegativity follows directly from $\sigma_i(k) \in [0, 1)$ and definition of normalized channel coefficients.

Row-stochasticity means that, $\forall i \in \mathcal{N}$,

$$(1-\sigma_i(k)) + \sigma_i(k)\sum_{j=1}^{n} h_{ij}(k) = 1. \tag{5.15}$$

For agent $i \in \mathcal{N}$ such that $|\mathcal{N}_i(k)| = 0$, by (5.11), equation (5.15) is trivially verified. Otherwise, having agent $i \in \mathcal{N}$ at least one neighbor, by Observation 1, $\sum_{j=1}^{n} h_{ij}(k) = 1$, which yields

$$(1-\sigma_i(k)) + \sigma_i(k)\sum_{j=1}^{n} h_{ij}(k) = (1-\sigma_i(k)) + \sigma_i(k) = 1.$$

Therefore, (5.15) is verified and the proof is concluded. □

Observation 2. *In general, $D_n^{\sigma}(k)$ is neither symmetric nor column-stochastic.*

Proposition 19. *If the underlying network topology $\mathfrak{G}$ is a sequence of strongly connected graphs, $\{D_n^{\sigma}(k)\}_{k\in\mathbb{N}_0}$ is a sequence of row-stochastic primitive matrices.*

Proof. Let $\mathfrak{G}$ be a sequence of strongly connected graphs $\mathcal{G}(k)$, $k \in \mathbb{N}_0$. By Proposition 18, $D_n^{\sigma}(k)$ is row-stochastic by construction. By definition of normalized channel coefficients, $\forall k \in \mathbb{N}_0$, each $h_{ij}(k)$ is positive if (and only if) there is an arc in $\mathcal{G}(k)$ from j to i. Moreover,

strongly connectedness implies $|\mathcal{N}_i(k)| > 0$, $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$. Therefore, $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $\sigma_i(k) > 0$. As a consequence, strongly connectedness of $\mathcal{G}(k)$ is equivalent to matrix

$$A_n^\sigma(k) = \begin{bmatrix} 0 & \sigma_1(k)h_{12}(k) & \dots & \sigma_1(k)h_{1n}(k) \\ \sigma_2(k)h_{21}(k) & 0 & \dots & \sigma_2(k)h_{2n}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_n(k)h_{n1}(k) & \sigma_n(k)h_{n2}(k) & \dots & 0 \end{bmatrix}.$$

being irreducible (see [37, Theorem 6.2.24]). Note that

$$\forall k \in \mathbb{N}_0,\ D_n^\sigma(k) = A_n^\sigma(k) + \Sigma(k),$$

where $\Sigma(k) := \text{diag}(1 - \sigma_1(k), \dots, 1 - \sigma_n(k))$ is also irreducible (see Definition 7). By [37, Lemma 8.5.4], an irreducible matrix with positive diagonal entries is primitive, thus $D_n^\sigma(k)$ is primitive, $\forall k \in \mathbb{N}_0$. This concludes the proof. □

5.1.3.1 *Time-invariant case*

The case in which both network topology and channel coefficients do not vary over $k \in \mathbb{N}_0$, is of particular interest, since the agreement value can be explicitly calculated. Constant channel coefficients means that, $\forall k \in \mathbb{N}_0$, $\forall i, j \in \mathcal{N}$, $h_{ij}(k) = h_{ij}$. As a consequence, $\forall k \in \mathbb{N}_0$, $\mathcal{G}(k) = \mathcal{G}$. We also assume constant stubbornness parameters. By these considerations, D_n^σ becomes

$$D_n^\sigma := \begin{bmatrix} 1 - \sigma_1 & \sigma_1 h_{12} & \dots & \sigma_1 h_{1n} \\ \sigma_2 h_{21} & 1 - \sigma_2 & \dots & \sigma_2 h_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_n h_{n1} & \sigma_n h_{n2} & \dots & 1 - \sigma_n \end{bmatrix}, \tag{5.16}$$

so that

$$\boldsymbol{x}(k+1) = D_n^\sigma \boldsymbol{x}(k). \tag{5.17}$$

The following lemma states that the power of D_n^σ converges.

Lemma 1. *If $\mathcal{G}$ is strongly connected, $\lim_{k\to\infty} (D_n^\sigma)^k = \mathbf{1}_n \boldsymbol{w}_n{}'$, $\boldsymbol{w}_n{}' \in \mathbb{R}^n_{>0}$. Moreover, $\boldsymbol{w}_n{}' D_n^\sigma = \boldsymbol{w}_n{}'$.*

Proof. By Proposition 19, $\mathcal{G}$ being strongly connected, D_n^σ is primitive and row-stochastic. By Wielandt's theorem (see Theorem 1),

$$\bar{D_n^\sigma} := (D_n^\sigma)^{n^2-2n+2} > 0.$$

Trivially,

$$\lim_{k\to\infty} (D_n^\sigma)^k = \lim_{k\to\infty} (\bar{D_n^\sigma})^{\frac{k}{n^2-2n+2}} = \lim_{k\to\infty} (\bar{D_n^\sigma})^k,$$

which is an infinite product of positive matrices, a topic with a long tradition in matrix theory, see [31] and [37]. Matrix $\bar{D_n^\sigma}$ is also row-stochastic, since product of row-stochastic matrices (see Proposition 2).

$\lambda = 1$ is an eigenvalue of matrix $\bar{D}_n^\sigma$; this follows directly from $\bar{D}_n^\sigma$ being row-stochastic, i.e.,

$$\bar{D}_n^\sigma \mathbf{1}_n = 1 \cdot \mathbf{1}_n.$$

By the Gershgorin disc theorem (see Theorem 3), there is no eigenvalue which is larger than $\lambda = 1$ in magnitude, since the largest Gershgorin disc is the one centered in the origin with radius 1. Whether this eigenvalue is simple (unique) needs to be determined by using the Perron-Frobenius Theorem. In fact, by the Perron-Frobenius Theorem, (see Theorem 2), for $\bar{D}_n^\sigma$ positive matrix, the largest eigenvalue (i.e., $\lambda = 1$) is also its spectral radius and all other eigenvalues are smaller in modulus. This yields that $\lambda = 1$ is the largest eigenvalue and it is unique. By the same theorem, $\boldsymbol{w}_n' \bar{D}_n^\sigma = \rho(\bar{D}_n^\sigma)\boldsymbol{w}_n' = \boldsymbol{w}_n'$, where $\boldsymbol{w}_n' \mathbf{1}_n = 1$, and

$$\lim_{k\to\infty} (\rho(\bar{D}_n^\sigma)^{-1} \bar{D}_n^\sigma)^k = \lim_{k\to\infty} (\bar{D}_n^\sigma)^k = \mathbf{1}_n \boldsymbol{w}_n'.$$

This leads to obtaining

$$\lim_{k\to\infty} (D_n^\sigma)^k = \lim_{k\to\infty} (\bar{D}_n^\sigma)^k = \mathbf{1}_n \boldsymbol{w}_n',$$

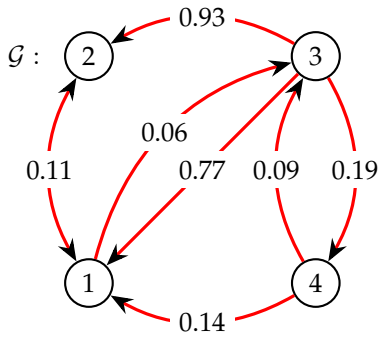
which concludes the proof. □

Although entries of $\lim_{k\to\infty} (D_n^\sigma)^k$ do depend on the unknown channel coefficients and cannot be determined a priori, some interesting properties of the consensus value can be investigated.

Proposition 20. *If $\mathcal{G}$ is strongly connected, system (5.17) achieves consensus asymptotically, i.e., $\exists x^* \in \mathbb{R}$, such that*

$$\lim_{k\to\infty} \boldsymbol{x}(k) = \boldsymbol{x}^* := x^* \mathbf{1}_n \tag{5.18}$$

Proof. By Lemma 1, $\lim_{k\to\infty} \boldsymbol{x}(k) = \lim_{k\to\infty} (D_n^\sigma)^k \boldsymbol{x}(0) = \mathbf{1}_n \boldsymbol{w}_n' \boldsymbol{x}(0)$. By the latter, it is immediate noting that $x^* = \boldsymbol{w}_n' \boldsymbol{x}(0)$, thus confirming that the agreement value does depend on the unknown channel coefficients. This concludes the proof. □



$$\mathfrak{A} = \begin{bmatrix} 0 & 0 & 0.77 & 0.14 \\ 0.11 & 0 & 0.93 & 0 \\ 0.06 & 0.11 & 0 & 0.09 \\ 0 & 0 & 0.19 & 0 \end{bmatrix}$$

Figure 5.2: Network topology and corresponding adjacency matrix for Example 12.

Example 12. *A system composed of $n = 4$ agents is simulated, with $\sigma_i = \sigma = 0.4$. The underlying network topology and its corresponding adjacency matrix are in Figure 1, where arc weights are determined by the corresponding channel coefficient. Channel coefficients are randomly generated out of a uniform distribution. Also the initial information states vector is as in that example, i.e.,*

$$\boldsymbol{x}(0) = [0.5, 2, -1, 0.1]'.$$

Figure 5.3 illustrates the results, thus showing that consensus is achieved. However, the agreement value is not the linear average of initial states, rather a weighted average of them.

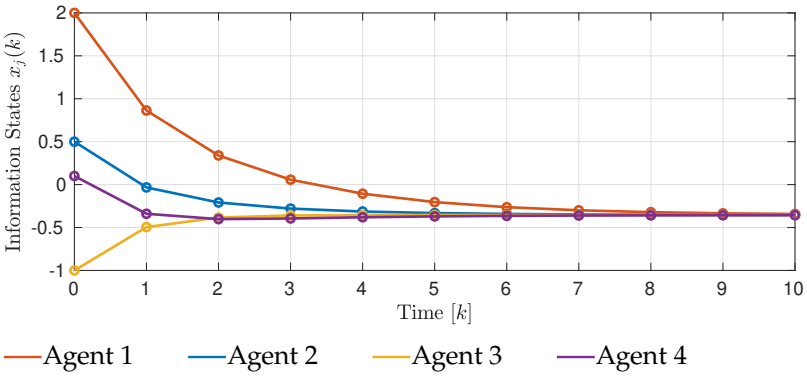


Figure 5.3: Simulation for Example 12.

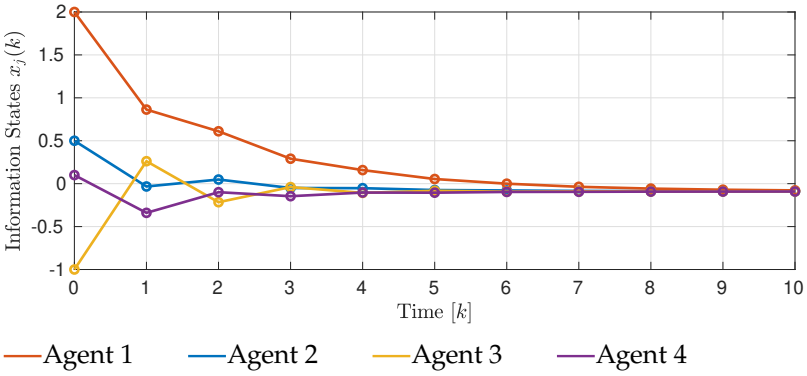


Figure 5.4: Simulation for Example 12.

If σ_3 is now set to be much larger than other agents', e.g., $\sigma_3 = 0.9$, (which results in a lower stubborness for agent 3) for the same channel realization as before, we obtain a different x^*, not as close as $x_3(0)$ as previously, see Figure 5.4.

Corollary 2. $x^* \in \mathfrak{C}\left(\{x_i(0)\}_{i\in\mathcal{N}}\right)$.

Proof. By Proposition 20, $x^* = \boldsymbol{w}_n'\boldsymbol{x}(0)$. By the proof of Lemma 1, $\boldsymbol{w}_n'\mathbf{1}_n = 1$. Thus, $x^* = \sum_{i=1}^n [\boldsymbol{w}_n]_i x_i(0)$, $\sum_{i=1}^n [\boldsymbol{w}_n]_i = 1$, proving that x^* is a convex combination of all initial information states. This concludes the proof. □

Corollary 3. *If, $\forall i \in \mathcal{N}$, $\sigma_i = \sigma$, then x^* does not depend on σ.*

Proof. $\boldsymbol{w}_n'$ is the left eigenvector of $(D_n^\sigma - \mathbb{I}_n)$ to $\lambda = 0$, i.e.

$$\boldsymbol{w}_n'(D_n^\sigma - \mathbb{I}_n) = \boldsymbol{w}_n' \cdot 0.$$

Note that, as shown earlier, $\boldsymbol{w}'_n$ is also the left eigenvector of D_n^σ to $\lambda = 1$.[3] This latter implies

$$\boldsymbol{w}'_n \cdot \sigma \begin{bmatrix} -1 & h_{12} & \dots & h_{1n} \\ h_{21} & -1 & \dots & h_{2n} \\ \dots & \dots & \dots & \dots \\ h_{n1} & h_{n2} & \dots & -1 \end{bmatrix} = \mathbf{0}_n, \tag{5.19}$$

thus showing that the entries of $\boldsymbol{w}'_n$ do not depend on the chosen σ.[4] Since x^* depends only on $\boldsymbol{w}_n$ and $\boldsymbol{x}(0)$, both not depending on σ, it does not depend on σ. This yields the proof. □

[3] $\boldsymbol{w}'_n D_n^\sigma = \boldsymbol{w}'_n$ implies $\boldsymbol{w}'_n D_n^\sigma - \boldsymbol{w}'_n = \boldsymbol{w}'_n (D_n^\sigma - \mathbb{I}_n) = \mathbf{0}_n$.

[4] For getting to this result, it just suffices to multiply by $\frac{1}{\sigma}$ both sides of (5.19).

Corollary 4. *If, $\forall i, j \in \mathcal{N}$, $h_{ij}(k) = h_{ji}(k)$, then $x^* = \frac{1}{n}\sum_{i=1}^n x_i(0)$.*

Proof. $\forall i, j \in \mathcal{N}$, $h_{ij}(k) = h_{ji}(k)$ implies $D_n^\sigma = (D_n^\sigma)'$. This yields $(\frac{1}{n}\mathbf{1}'_n)D_n^\sigma = (\frac{1}{n}\mathbf{1}'_n)$, thus $\lim_{k\to\infty}(D_n^\sigma)^k = \frac{1}{n}\mathbf{1}_n\mathbf{1}'_n$. The latter implies $x^* = \frac{1}{n}\mathbf{1}'_n\boldsymbol{x}(0) = \frac{1}{n}\sum_{i=1}^n x_i(0)$, thus concluding the proof. □

Example 13. *With regards to Figure 5.3 in Example 12, we simulate the same system (with identical channel coefficients and initial states vector). Figure 5.5 illustrates what happens to the system when $\sigma = 0.1$. The convergence speed slows down significantly, but x^* is the same. The same occurs also for $\sigma = 0.7$, for which, conversely, converging speed increases, see Figure 5.6. On the other hand, Figure 5.7 shows that increasing σ too much (in this case, $\sigma = 0.9$) results in big oscillations and slower convergence.*

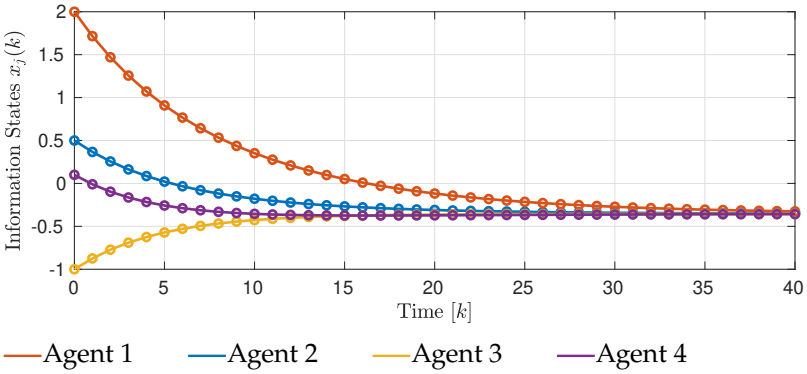


Figure 5.5: Simulation for Example 13 with $\sigma = 0.1$. Note that the x-axis shows up to $k = 40$, which is 4 times larger than its equivalent in Figure 5.3. This proves the slower convergence.

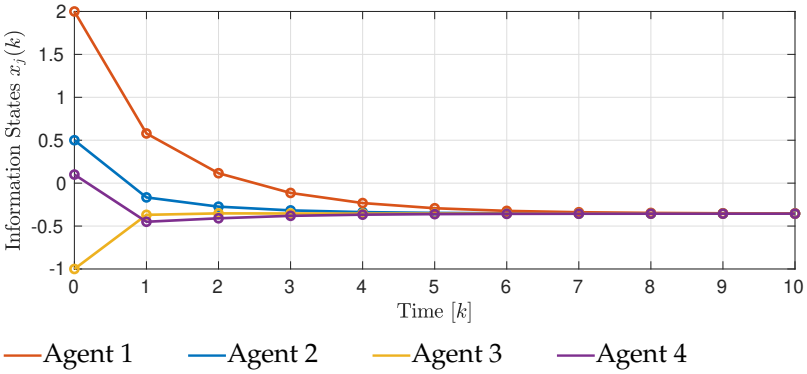


Figure 5.6: Simulation for Example 13 with $\sigma = 0.7$.

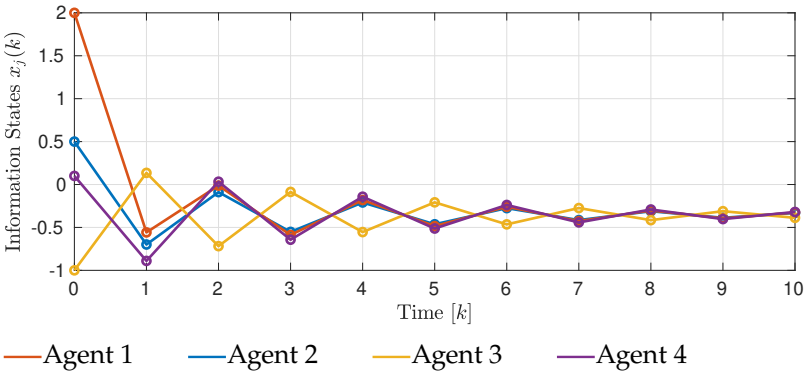


Figure 5.7: Simulation for Example 13 with $\sigma = 0.95$.

Example 13 leads us to talking about the convergence rate of system (5.17). The topic has been also addressed in literature as *performance* of consensus. However, all relative contributions (see, e.g., [20, 80, 94]) consider either undirected network topologies or balanced directed topologies.[5] Such a graphical property is far from reality in our system's setting. In fact, besides the network structure being directed, Observation 2 points out that D_n^σ is in general not column-stochastic, which translates in having an unbalanced underlying topology. In what follows, we present an analysis of convergence rate of system (5.17) in the case the underlying topology is directed and unbalanced. To the best of our knowledge, this constitutes a novelty. Only [106] considers a similar problem, but presents the results only for D_n^σ being diagonalizable. This latter condition is relaxed in what follows.

[5] See Definition 10

Theorem 7. *If $\mathcal{G}$ is strongly connected, system (5.17) achieves consensus asymptotically with a rate of convergence determined by the eigenvalue of D_n^σ with the second largest magnitude.*

Proof. Let $\lambda_1, \dots, \lambda_n$ with

$$|\lambda_1| \leq \cdots \leq |\lambda_{n-1}| < \lambda_n = 1$$

be the eigenvalues of D_n^σ. The largest eigenvalue λ_n has already been shown to be real-valued unique and equal to 1 in the proof of Lemma 1 (given that $\mathcal{G}$ is strongly connected). All other eigenvalues are smaller in modulus and some of them can be mutually equal. In this case, D_n^σ could be non-diagonalizable. This constitutes a more general case than what considered in [106]. Let

$$D_n^\sigma = QJQ^{-1},$$

where Q is a square $n \times n$ matrix with columns composed of n linearly independent generalized eigenvectors $\boldsymbol{v}_i$, $i = 1, \dots, n$, and J is the *Jordan normal form* of D_n^σ. See Appendix A.1 for a discussion about generalized eigenvectors and the Jordan normal form. Let's now focus on the vector of initial information states, i.e., $\boldsymbol{x}(0) \in \mathbb{R}^n$. Every vector in $\mathbb{R}^n$ can be seen as a linear combination of n linearly independent vectors of $\mathbb{C}^n$. In case these linearly independent vectors are chosen to be the n generalized eigenvectors $\boldsymbol{v}_i$, we obtain

$$\boldsymbol{x}(0) = \sum_{i=1}^{n} \gamma_i \boldsymbol{v}_i, \tag{5.20}$$

for some $\gamma_i \in \mathbb{C}$, $i = 1, \dots, n$. By this, it follows that, $\forall k \in \mathbb{N}_0$,

$$\begin{aligned} \boldsymbol{x}(k) &= (D_n^\sigma)^k \boldsymbol{x}(0) \\ &= (D_n^\sigma)^k \sum_{i=1}^{n} \gamma_i \boldsymbol{v}_i = \sum_{i=1}^{n} \gamma_i (D_n^\sigma)^k \boldsymbol{v}_i. \end{aligned} \tag{5.21}$$

Since, $\forall i = 1 \dots n$, $\boldsymbol{v}_i$ is a generalized eigenvector of $(D_n^\sigma)^k = QJ^kQ^{-1}$, we can employ the result of Theorem A.1.4 in Appendix A.1. This leads to having, $\forall k \in \mathbb{N}_0$,

$$\boldsymbol{x}(k) = \sum_{i=1}^{n} \gamma_i \left(\sum_{\ell=0}^{o(i)-1} \binom{k}{\ell} \lambda_i^{k-\ell} \boldsymbol{v}_i \right), \tag{5.22}$$

where $o(i)$ is the rank of the i-th generalized eigenvector. Equation 5.22 shows that, for $k \to \infty$, one component (the one relative to $\lambda_n = 1$) is converging to consensus (in this case, $\boldsymbol{x}^* = \gamma_n \mathbf{1}_n$). All other contributions converge to 0, as

$$\forall i = 1, \dots, n-1, \ |\lambda_i| < 1 \implies \lim_{k \to \infty} |\lambda_i|^{k-\ell} = 0,$$

for each $\ell = 0, \dots, o(i)$. The rate of $\boldsymbol{x}(k)$ converging to the consensus is determined by the second largest eigenvalue (simple or not) λ_{n-1}, since its corresponding element in (5.22) converges more slowly than all other other terms. In case λ_{n-1} and λ_{n-2} are complex conjugate eigenvalues, then the rate of convergence will be determined by $|\lambda_{n-1}| = |\lambda_{n-2}|$. This concludes the proof. □

Following the outcome of Example 13, we aim at characterizing the value of σ for which the convergence rate is maximized.

Lemma 2. *Let the underlying network topology be strongly connected. If, $\forall i \in \mathcal{N}$, $\sigma_i = \sigma$, then the convergence rate depends on the choice of σ and on the eigenvalues of $\mathfrak{A}$, where $\mathfrak{A}$ is matrix of normalized channel coefficients, i.e.,*

$$\mathfrak{A} = \begin{bmatrix} 0 & h_{12} & \dots & h_{1n} \\ h_{21} & 0 & \dots & h_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n1} & h_{n2} & \dots & 0 \end{bmatrix}.$$

Proof. Matrix D_n^σ can be rewritten as

$$D_n^\sigma = \Sigma + \sigma \mathfrak{A},$$

where $\Sigma = (1 - \sigma)\mathbb{I}_n$. Note that Σ and $\sigma\mathfrak{A}$ commute. By this,

$$\text{eig}\,(D_n^\sigma) = \text{eig}\,(\Sigma) + \sigma\,\text{eig}\,(\mathfrak{A}) = 1 - \sigma + \sigma\,\text{eig}\,(\mathfrak{A}). \tag{5.23}$$

By the Geršgorin disc theorem, all eigenvalues of $\mathfrak{A}$ are within or on the unit circle. Matrix $\mathfrak{A}$ is irreducible, since the underlying network topology is strongly connected. By the Perron-Frobenius theorem for irreducible matrices, see, e.g., [31, pg. 182], $\text{eig}\,(\mathfrak{A}) = 1$ is simple and dominates in modulus all other eigenvalues. The eigenvalue of D_n^σ corresponding to $\text{eig}\,(\mathfrak{A}) = 1$ is also 1. By Theorem 7, and by

$\lambda_{n-1}^{\sigma} \in \mathbb{C}$ being the second largest eigenvalue in modulus of D_n^{σ}, the convergence rate of the considered problems depends on $|\lambda_{n-1}^{\sigma}|$. By (5.23), and for a given $\sigma \in (0,1)$,

$$|\lambda_{n-1}^{\sigma}| = \max_{\lambda \in \text{eig}(\mathfrak{A}),\ \lambda \neq 1} |1 - \sigma + \sigma\lambda|. \tag{5.24}$$

This concludes the proof. □

The analysis of the system for the case of different stubbornness parameters is not straightforward and relies on a lower bound condition for the second largest eigenvalue, which comes directly from the Geršgorin Disc Theorem.

Proposition 21. *If $\mathcal{G}$ is strongly connected,*

$$\forall i = 1, \dots, n,\ \sigma_i \leq \tilde{\sigma} \implies |\lambda_{n-1}| \geq 1 - 2\tilde{\sigma}.$$

Proof. This proof is based on the Geršgorin disc theorem, see Theorem 3. If $\forall i = 1, \dots, n,\ \sigma_i \leq \tilde{\sigma}$, then all Geršgorin discs are contained in the disc of radius $\tilde{\sigma}$ centered in $1 - \tilde{\sigma}$, see Figure 5.8. All eigenvalues are contained in the same disc of radius $\tilde{\sigma}$ centered in $1 - \tilde{\sigma}$, with $\tilde{\sigma} \in (0,1)$ by definition. The point with minimum modulus belonging to this circle is on the x-axis and has abscissa $1 - 2\tilde{\sigma}$. This concludes the proof. □

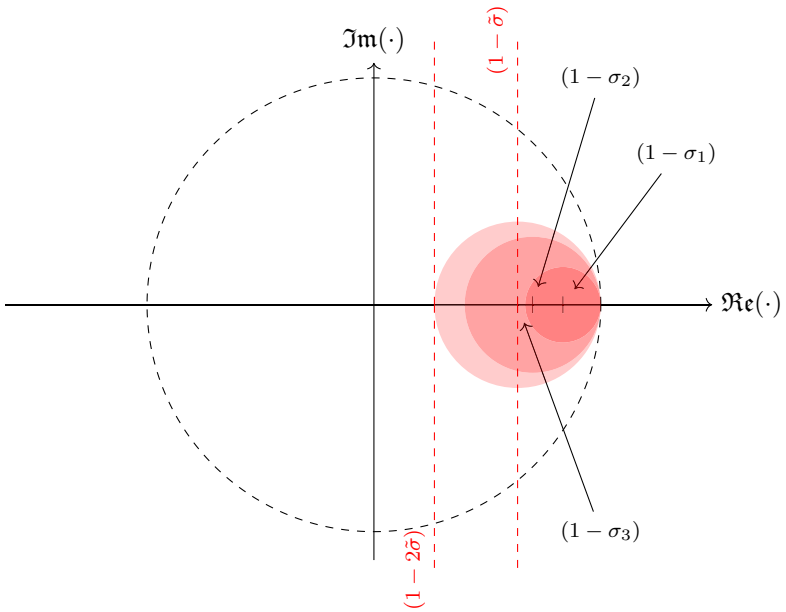


Figure 5.8: Geršgorin discs for a square matrix of dimension 3. All eigenvalues are contained in the union of discs.

Proposition 21 implies that, if all σ_i are small, convergence will be slow.

5.1.3.2 Time-variant case

By (5.13),

$$\lim_{k\to\infty} \boldsymbol{x}(k) = \lim_{k\to\infty} D_n^\sigma(k-1)D_n^\sigma(k-2)\dots D_n^\sigma(1)D_n^\sigma(0)\boldsymbol{x}(0). \quad (5.25)$$

Indeed, convergence to a consensus follows from the infinite product of nonnegative row-stochastic matrices $\lim_{k\to\infty} D_n^\sigma(k-1)\dots D_n^\sigma(0)$. Concerning convergence of infinite products of row-stochastic matrices, the main literature result can be found in [113] and has been reviewed in Section 2.2.

Sequence of strongly connected topologies

Let $\mathfrak{G}$ be the sequence of network topologies. We prove the following result.

Proposition 22. *Given a sequence of strongly connected network topologies $\mathfrak{G}$ and time-varying (unknown) channel coefficients, system (5.13) achieves consensus asymptotically for any choice of the stubbornness parameters.*

Proof. Independently of the choice of the $\sigma_i(k)$, $\forall k \in \mathbb{N}_0$, Proposition 19 states that $D_n^\sigma(k)$ is row-stochastic and primitive. To analyze

$$\lim_{k\to\infty} D_n^\sigma(k-1)D_n^\sigma(k-2)\dots D_n^\sigma(1)D_n^\sigma(0),$$

we need to review Theorem 4 and (2.3). By these, $\exists \boldsymbol{r}_n \in \mathbb{R}^n$, with $\sum_{i=1}^n [\boldsymbol{r}_n]_i = 1$, such that $\lim_{k\to\infty} D_n^\sigma(k-1)D_n^\sigma(k-2)\dots D_n^\sigma(1)D_n^\sigma(0) = \mathbf{1}_n\boldsymbol{r}_n'$. By (5.25), $\lim_{k\to\infty} \boldsymbol{x}(k) = \mathbf{1}_n\boldsymbol{r}_n'\boldsymbol{x}(0)$, thus yielding that the agreement value is $x^* = \boldsymbol{r}_n'\boldsymbol{x}(0)$. The proof is concluded. □

Since $\boldsymbol{r}_n$ is such that $\sum_{i=1}^n [\boldsymbol{r}_n]_i = 1$, Corollary 2 applies also to the time-variant case and $x^* \in \mathfrak{C}(\{x_i(0)\}_{i\in\mathcal{N}})$. Clearly, this value does depend on the unknown channel coefficients (and on the sequence of network topologies) and its exact value cannot be determined a priori.

Example 14. *A system, whose underlying network topology consists of the four graphs in Figure 5.9 sequentially repeated, exhibits dynamics (5.13). We run a first simulation in Figure 5.10. By changing the fading channel coefficients, as in Figure 5.11, also the agreement value varies. On the other hand, by keeping the fading channel coefficients of the first simulation, we can compare the impact of σ. In Figure 5.12 we illustrate the implications of a smaller σ ($\sigma = 0.2$). Convergence rate is much slower. Converging rate increases in Figure 5.13, whereas $\sigma = 0.6$. Figure 5.14 shows the impact of an even higher σ, which leads to an extremely oscillating behavior.*

As for the time-invariant case, concerning the convergence rate, literature mostly focuses on the case of undirected (or balanced directed) underlying network topologies, see, e.g., [78]. However, the proposed setting (balanced directed network topology) is not compatible with the presence of an unknown (non-symmetric) fading channel. A more general analysis that considers directed unbalanced graphs is, instead, proposed in [80]. Following this strategy, we employ the

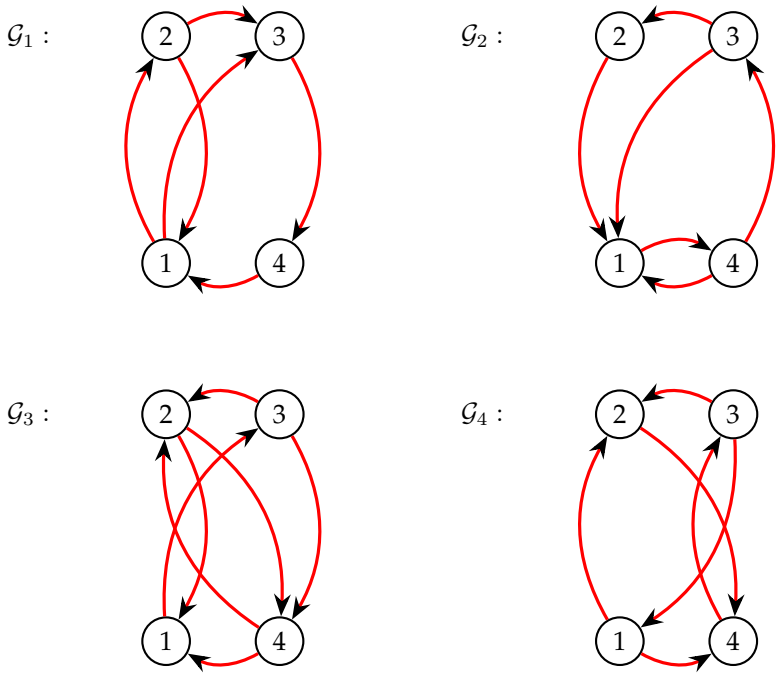


Figure 5.9: Underlying network topology for Example 14.

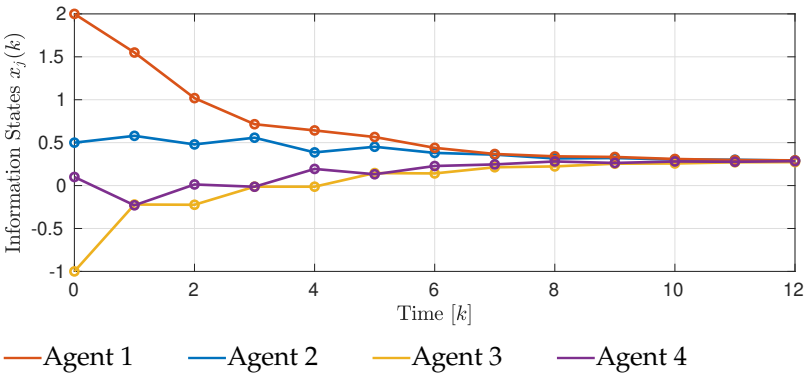


Figure 5.10: System (5.13) with random channel and $\sigma = 0.3$

so-called *nullifying matrix* $U \in \mathbb{R}^{(n-1)\times n}$ that is thoroughly analyzed in Appendix A.2. Matrix U is a full-rank projection matrix from $\mathbb{R}^n$ onto $\mathbb{R}^{n-1}$, i.e., $U \in \mathbb{R}^{(n-1)\times n}$, such that

- $U\mathbf{1}_n = \mathbf{0}_{n-1}$;[6]
- $UU' = \mathbb{I}_{n-1}$;[7]
- $U'U = \mathbb{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n'$;[8]
- $\|U\| = \|U'\| = 1$;[9]
- by Proposition A.2.2, $\forall k \in \mathbb{N}_0$,[10]

$$\exists!\, Q^\sigma(k) \in \mathbb{R}^{(n-1)\times(n-1)} : \; U D_n^\sigma(k) = Q^\sigma(k) U$$

and

$$\text{spectrum}\,(D_n^\sigma(k)) = \{1\} \cup \text{spectrum}\,(Q^\sigma(k))\,.$$

It comes as a consequence that

$$\forall k \in \mathbb{N}_0,\ \rho(Q^\sigma(k)) = |\lambda_{n-1}(D_n^\sigma(k))|, \tag{5.26}$$

[6] See (A.12).

[7] See (A.13).

[8] See Theorem A.2.1.

[9] See Proposition A.2.1.

[10] $\exists!$ means that exists and is unique.

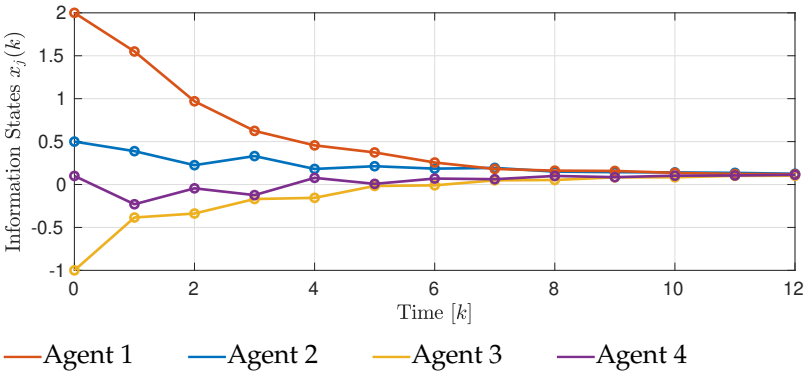


Figure 5.11: System (5.13) as in Figure 5.10, but another channel realization.

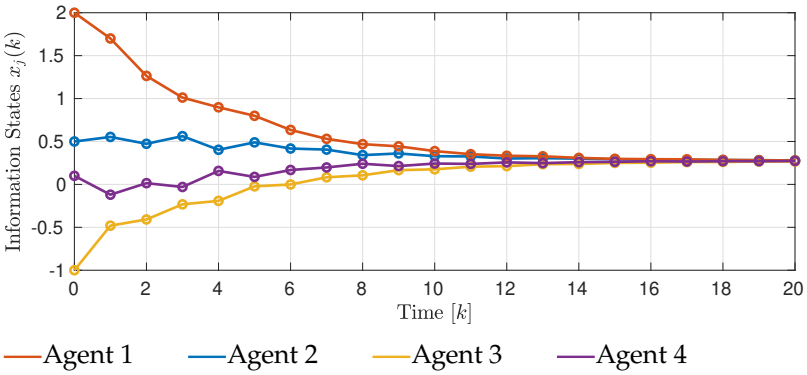


Figure 5.12: System (5.13) as in Figure 5.10, but $\sigma = 0.2$.

namely, the spectral radius of $Q^\sigma(k)$ is equal to the modulus of the second largest eigenvalue of $D_n^\sigma(k)$. These considerations are of use for Proposition 23, that links the convergence rate of system (5.13) to the joint spectral radius of the matrix sequence $\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}$.

Definition 15 (Joint Spectral Radius, *Protasov* [84])**.** *Let* $\mathbf{M}$ *be a finite set of* $n \times n$ *real valued matrices, i.e.,* $\mathbf{M} = \{M(1), \dots, M(R)\}$*, with* R *being the cardinality of* $\mathbf{M}$*. Then, the joint spectral radius of* $\mathbf{M}$ *is*

$$\rho(\mathbf{M}) := \lim_{k\to\infty} \max_{\varrho} \|M(\varrho(1)) \cdot \dots \cdot M(\varrho(k))\|, \quad \varrho : \{1, \dots, k\} \to \{1, \dots, R\}. \tag{5.27}$$

Assumption 5. *To meaningfully estimate the convergence rate, we assume that fading channel coefficients are quantized, i.e.,*

$$\xi_{ij}(k) : \mathbb{R}_{\geq 0} \mapsto S \subset \mathbb{R},$$

$\forall i \in \mathcal{N}$, $\forall j \in \mathcal{N}_i(k)$*, with* S *being a finite set. This implies that* $\{D_n^\sigma(k)\}_{k\in\mathbb{N}_0}$ *is arbitrarily large, but finite. Under this assumption, also* $\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}$ *is a finite set.*

Lemma 3. *If* $\boldsymbol{x}_n \in \mathbb{R}^n$ *is such that* $\boldsymbol{x}_n' \mathbf{1}_n = 0$*, then*

$$\|U\boldsymbol{x}_n\| = \|\boldsymbol{x}_n\|.$$

Proof. By definition of 2-norm and by Theorem A.2.1,

$$\|U\boldsymbol{x}_n\|^2 = \boldsymbol{x}_n' U' U \boldsymbol{x}_n = \boldsymbol{x}_n' (\mathbb{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n')\boldsymbol{x}_n.$$

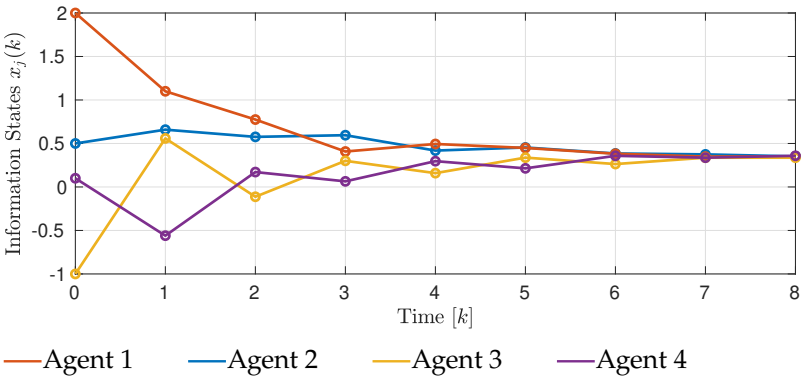


Figure 5.13: System (5.13) as in Figure 5.10, but $\sigma = 0.6$.

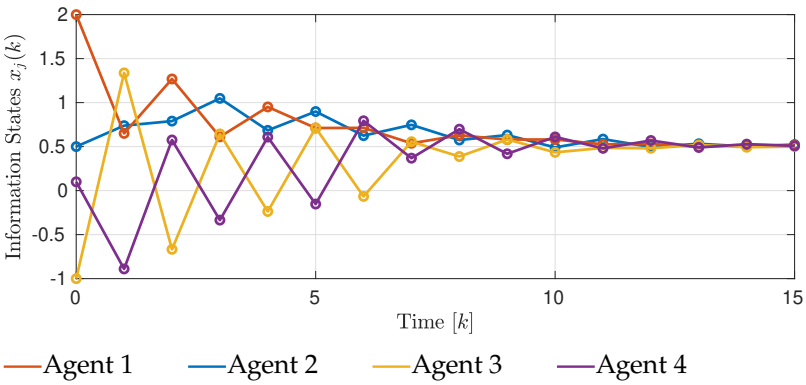


Figure 5.14: System (5.13) as in Figure 5.10, but $\sigma = 0.9$.

By hypothesis of the theorem,

$$\boldsymbol{x}_n'(\frac{1}{n}\mathbf{1}_n\mathbf{1}_n')\boldsymbol{x}_n = 0,$$

thus,

$$\|U\boldsymbol{x}_n\|^2 = \boldsymbol{x}_n'(\mathbb{I}_n)\boldsymbol{x}_n = \|\boldsymbol{x}_n\|^2.$$

□

Proposition 23. *The convergence rate of system (5.13) depends on the joint spectral radius of* $\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}$.

Proof. Let $\forall k \in \mathbb{N}_0$, $\gamma(k)$ be defined as

$$\gamma(k) := \frac{1}{n}\mathbf{1}_n'\boldsymbol{x}(k).$$

Note that $U(\gamma(k)\mathbf{1}_n) = \mathbf{0}_{n-1}$. We also have

$$U(\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n) = U\boldsymbol{x}(k) - U\gamma(k)\mathbf{1}_n = U\boldsymbol{x}(k). \tag{5.28}$$

Indeed, by Proposition A.2.2, $\forall k \in \mathbb{N}_0$,

$$\begin{aligned} U\boldsymbol{x}(k+1) &= UD_n^\sigma(k)\cdot\cdots\cdot D_n^\sigma(0)\boldsymbol{x}(0) \\ &= Q^\sigma(k)\cdot\cdots\cdot Q^\sigma(0)U\boldsymbol{x}(0). \end{aligned} \tag{5.29}$$

By merging (5.28) and (5.29), one obtains

$$U(\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n) = Q^\sigma(k-1)\cdot\cdots\cdot Q^\sigma(0)U\boldsymbol{x}(0). \tag{5.30}$$

Note that $(\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n)'\mathbf{1}_n = 0$, thus, by Lemma 3,

$$\|U(\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n)\| = \|\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n\|.$$

By merging the latter together with (5.30), we obtain

$$\begin{aligned}\|\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n\| &= \|Q^\sigma(k-1)\cdot\dots\cdot Q^\sigma(0)U\boldsymbol{x}(0)\| \\ &\leq \|Q^\sigma(k-1)\cdot\dots\cdot Q^\sigma(0)\|\,\|U\|\,\|\boldsymbol{x}(0)\|. \end{aligned} \tag{5.31}$$

By Proposition A.2.1, $\|U\| = 1$. Using the concept of joint spectral radius (see Definition 15), and by [13, Sec. IV], for any $q > \rho(\{Q^\sigma(k)\}_{k\in\mathbb{N}_0})$, there exists a $C \in \mathbb{R}$, such that

$$\|\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n\|_2 \leq Cq^k\,\|\boldsymbol{x}(0)\|_2. \tag{5.32}$$

Note that $x^* \in \mathfrak{C}(\{\boldsymbol{x}(k)\}_{k\in\mathbb{N}_0})$, $\forall k \in \mathbb{N}_0$, thus

$$\|\boldsymbol{x}(k) - x^*\mathbf{1}_n\|_\infty \leq 2\,\|\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n\|_\infty \leq 2\,\|\boldsymbol{x}(k) - \gamma(k)\mathbf{1}_n\|_2, \tag{5.33}$$

which finally leads to

$$\|\boldsymbol{x}(k) - x^*\mathbf{1}_n\|_\infty \leq 2Cq^k\,\|\boldsymbol{x}(0)\|_2. \tag{5.34}$$

Inequality (5.34) proves that the norm of the agreement error, defined as $(\boldsymbol{x}(k) - x^*\mathbf{1}_n)$, converges to 0 with the exponential of the joint spectral radius of $\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}$. This concludes the proof. □

Since the unknown fading channel changes at every iteration, obtaining also an exact value of σ that maximizes the convergence rate of the time-variant problem is, at the current state of the art, not possible. We can, instead, qualitatively explain the impact of σ for values either close to 0 or to 1:

- $\sigma \to 1$: all agents weigh the received information way more than their current value. This results in high oscillations (see Example 14, Figure 5.14);
- $\sigma \to 0$: all matrices $D_n^\sigma(k)$, $k \in \mathbb{N}_0$, are diagonally dominant; this results in a slower convergence, since the received value is heavily filtered out (see Example 14, Figure 5.12).

Jointly strongly connected sequence of network topologies

However, in literature, requiring a sequence of strongly connected network topologies is a condition that can still be relaxed. In fact, [88] showed that, in the traditional framework, consensus can be achieved asymptotically if the union of the underlying network topologies across some time interval is strongly connected[11] frequently enough. Such a graph sequence is called *repeatedly jointly strongly connected*.

[11] [88] does not mention connectivity, rather that the union of graphs should contain a spanning tree. However, as in Chapter 2, we do not consider this assumption to be a proper condition for consensus.

Lemma 4. *Integers $p, k \in \mathbb{N}_0$ and agents $i, j \in \mathcal{N}$ are given. If $\exists\{\ell_1, \dots, \ell_{p-1}\} \subseteq \mathcal{N} \setminus \{i, j\}$, such that*

$$h_{i\ell_{p-1}}(k+p)h_{\ell_{p-1}\ell_{p-2}}(k+p-1)\cdot\dots\cdot h_{\ell_1 j}(k+1) > 0,$$

then

$$[D_n^\sigma(k+p)\cdot\dots\cdot D_n^\sigma(k+1)]_{ij} > 0.$$

Proof.

$$\begin{aligned}
&[D_n^\sigma(k+p)\cdot\dots\cdot D_n^\sigma(k+1)]_{ij} = \\
&= \sum_{f=1}^{n}[D_n^\sigma(k+p)\cdot\dots\cdot D_n^\sigma(k+2)]_{if}[D_n^\sigma(k+1)]_{fj} \\
&\geq [D_n^\sigma(k+p)\cdot\dots\cdot D_n^\sigma(k+2)]_{i\ell_1}[D_n^\sigma(k+1)]_{\ell_1 j} \\
&= [D_n^\sigma(k+p)\cdot\dots\cdot D_n^\sigma(k+2)]_{i\ell_1}\sigma_{\ell_1}(k+1)h_{\ell_1 j}(k+1)
\end{aligned}$$

for $\ell_1 \in \mathcal{N}\setminus\{i,j\}$. By recursively expanding the product into the square brackets, one gets, $\forall\{\ell_1,\dots,\ell_{p-1}\}\subseteq\mathcal{N}\setminus\{i,j\}$,

$$\begin{aligned}
&[D_n^\sigma(k+p)\cdot\dots\cdot D_n^\sigma(k+1)]_{ij} \geq \\
&\qquad \geq \sigma_i(k+p)\sigma_{\ell_{p-1}}(k+p-1)\cdot\dots\cdot\sigma_{\ell_1}(k+1)\cdot\dots \\
&\qquad\quad \cdot\dots\cdot h_{i\ell_{p-1}}(k+p)h_{\ell_{p-1}\ell_{p-2}}(k+p-1)\cdot\dots\cdot h_{\ell_1 j}(k+1).
\end{aligned}$$

Being the stubbornness parameters always positive if the receiving agent has at least one neighbor (see (5.11)), it immediately follows that, $\forall p < n-1$,

$$\begin{aligned}
\exists\{\ell_1,\dots,\ell_{p-1}\}\subseteq\mathcal{N}\setminus\{i,j\} : h_{i\ell_{p-1}}(k+p)\cdot\dots\cdot h_{\ell_1 j}(k+1) > 0 \\
\implies [D_n^\sigma(k+p)\cdot\dots\cdot D_n^\sigma(k+1)]_{ij} > 0.
\end{aligned}$$

This concludes the proof. □

Proposition 24. *With a repeatedly jointly strongly connected sequence of underlying network topologies $\mathfrak{G}$ and time-varying channel coefficients, system (5.13) achieves consensus asymptotically for any choice of the stubbornness parameters.*

Proof. Since $\mathfrak{G} := \{\mathcal{G}(1),\mathcal{G}(2),\dots\}$ is a repeatedly jointly strongly connected sequence of graphs, there exists $p\in\mathbb{N}$ such that each finite sequence $\mathcal{G}(p(k+1))\circ\dots\circ\mathcal{G}(pk+1)$ is strongly connected. By definition of graph composition (see Section 2.1) and of normalized channel coefficients (see (5.8)), (i,j) is an arc of $\mathcal{G}(p(k+1))\circ\dots\circ\mathcal{G}(pk+1)$ iff

$$h_{i\ell_1}(pk+1)h_{\ell_1\ell_2}(pk+2)\cdot\dots\cdot h_{\ell_{p-1}j}(p(k+1))>0,$$

for some distinct $\ell_1,\dots,\ell_{p-1}\in\mathcal{N}$.
By Lemma 4, any off-diagonal entry (i,j) of $D_n^\sigma(p(k+1))\cdot\dots\cdot D_n^\sigma(pk+1)$ is positive if there is an arc from j to i in $\mathcal{G}(p(k+1))\circ\dots\circ\mathcal{G}(pk+1)$. This yields that, by [37, Theorem 6.2.24], if $\mathcal{G}(p(k+1))\circ\dots\circ\mathcal{G}(pk+1)$ is strongly connected, then $D_n^\sigma(p(k+1))\cdot\dots\cdot D_n^\sigma(pk+1)$ is irreducible. Moreover, all diagonal entries of $D_n^\sigma(p(k+1))\cdot\dots\cdot D_n^\sigma(pk+1)$ are always positive (trivial to prove). This yields, by [37, Lemma 8.5.4], that, if $\mathcal{G}(p(k+1))\circ\dots\circ\mathcal{G}(pk+1)$ is strongly connected, then $D_n^\sigma(p(k+1))\cdot\dots\cdot D_n^\sigma(pk+1)$ is primitive.
$\mathfrak{G}$ being repeatedly jointly strongly connected, then, every product of p consecutive primitive and row-stochastic matrices is primitive (and row-stochastic)[12], notwithstanding the unknown fading channel and

[12] Any matrix resulting from the product of row-stochastic matrices is also row-stochastic, see Proposition 2.

for every choice of the stubbornness parameters. By this,

$$\lim_{k\to\infty} D_n^\sigma(kp)D_n^\sigma(kp-1)\dots D_n^\sigma(1)D_n^\sigma(0)$$

is an infinite sequence of k primitive row-stochastic matrices, thus Proposition 22 can be employed and the proof concluded. □

Example 15. *In the spirit of Example 14, we simulate a system whose underlying topology is repeatedly jointly strongly connected, i.e., a sequence of the four graphs in Figure 5.15. In the simulations, we evaluate the impact of different policies for σ. In Figure 5.16, σ is chosen to be 0.4. The convergence speed can be diminished by picking $\sigma = 0.2$, as illustrated by Figure 5.17. As already shown for the cases of time-invariant and sequence of strongly connected topologies, picking a large σ, as in Figure 5.18 where it is chosen to be 0.9, leads to large oscillations, without improving the convergence speed.*

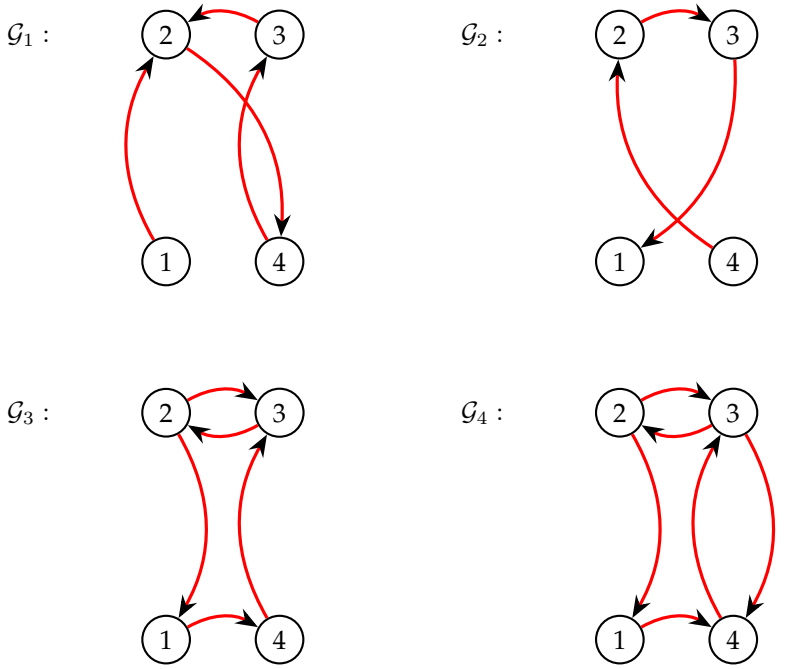


Figure 5.15: Underlying network topology for Example 15. The sequence of these four graphs constitutes the 4-repeatedly jointly strongly connected network topology.

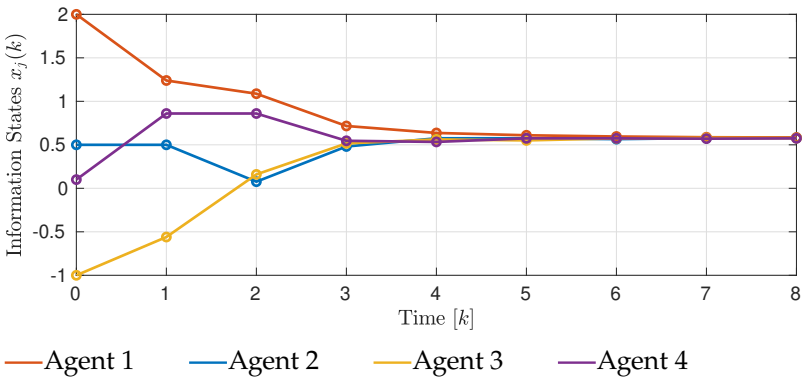


Figure 5.16: System 5.13 with the underlying network topology of Figure 5.15 and a random channel realization with $\sigma = 0.4$.

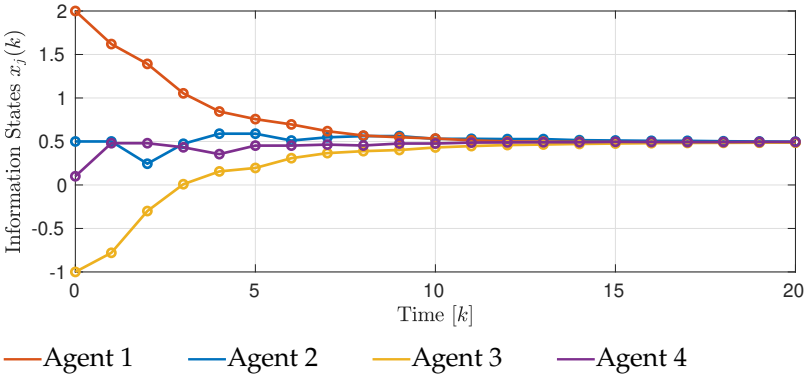


Figure 5.17: System as in Figure 5.16, but with $\sigma = 0.2$.

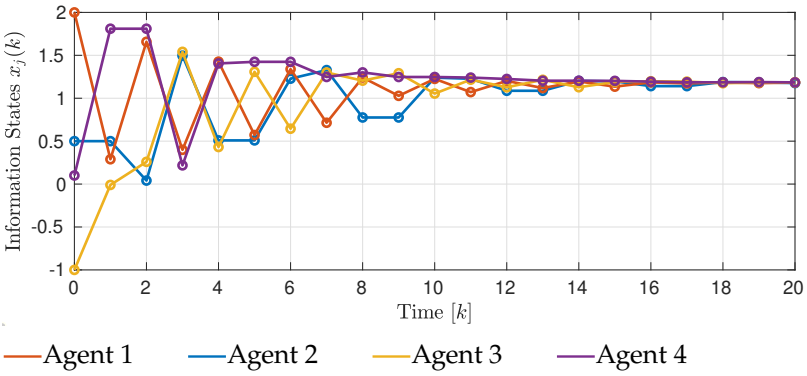


Figure 5.18: System as in Figure 5.16, but with $\sigma = 0.9$.

5.1.4 *Comparison with traditional approaches*

In this subsection, we compare our protocol to traditional approaches. By traditional approaches, as previously mentioned in this work, we denote those strategies that employ a standard consensus protocol as, e.g., [13, Sec. II] together with standard communication systems. In fact, each agent needs to reconstruct neighbors' states, thus orthogonal channel access methods[13] are employed. As explained elsewhere in this work and in literature, e.g., [34, 36, 107], the exploitation of the superposition property of the wireless channel leads to saving wireless resources, that, otherwise, would be employed for orthogonal transmissions.

[13] E.g., TDMA or FDMA.

Comparison in expectation.

As clarified in Chapter 4, channel fading coefficients are drawn out of an uniform distribution. We assume the following points:

- $\forall i, j, l \in \mathcal{N}, i \neq j \neq l, \forall k \in \mathbb{N}_0$, $\xi_{ij}(k)$ and $\xi_{il}(k)$ are independent;
- $\forall i, j \in \mathcal{N}, \forall k \in \mathbb{N}_0$, $\xi_{ij}(k)$ and $\xi_{ij}(k+1)$ are independent.

Concerning the general time-varying system (5.10), the fading channel distorts the signal carrying information about neighbors' states, i.e.,

$$\forall k \in \mathbb{N}_0, \qquad \zeta_i(k) := \frac{v_i(k)}{v_i'(k)} = \frac{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) x_j(k)}{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)}. \tag{5.35}$$

Proposition 25. *The expected value of $\zeta_i(k)$ (see 7.16) is the linear average of neighbors' states, i.e.,*

$$E\left[\zeta_i(k)\right] = \frac{\sum\limits_{j\in\mathcal{N}_i(k)} x_j(k)}{|\mathcal{N}_i(k)|}. \tag{5.36}$$

Proof. Let $\sum\limits_{j\in\mathcal{N}_i(k)} \xi_{ij}(k)$ be denoted by ϑ. Since channel coefficients are assumed to be i.i.d,

$$E[\vartheta] = \sum_{j\in\mathcal{N}_i(k)} E[\xi_{ij}(k)] = |\mathcal{N}_i(t_k)|E[\xi_{ij}].$$

By [27, Theorem 5.3.20],

$$\begin{aligned} E\left[\zeta_i(k)\right] &= E\left[\frac{\sum\limits_{j\in\mathcal{N}_i(k)} \xi_{ij}(k)x_j(k)}{\vartheta}\right] \\ &= \int_{-\infty}^{\infty} E\left[\frac{\sum\limits_{j\in\mathcal{N}_i(k)} \xi_{ij}(k)x_j(k)}{\vartheta}\middle|\vartheta = m\right] f_\vartheta(m)\mathrm{d}m, \end{aligned} \tag{5.37}$$

where f_ϑ is the probability density function of ϑ. By taking $\vartheta = m$ out of the expectation, one gets

$$E\left[\zeta_i(k)\right] = \int_{-\infty}^{\infty} \frac{1}{m} E\left[\sum_{j\in\mathcal{N}_i(k)} \xi_{ij}(k)x_j(k)\middle|\vartheta = m\right] f_\vartheta(m)\mathrm{d}m. \tag{5.38}$$

Since the expectation is a linear operator and channel coefficients are independent,

$$E\left[\zeta_i(k)\right] = \int_{-\infty}^{\infty} \frac{1}{m} \sum_{j\in\mathcal{N}_i(k)} E[\xi_{ij}(k) \mid \vartheta = m]x_j(k)f_\vartheta(m)\mathrm{d}m. \tag{5.39}$$

Let's focus now on $E[\xi_{ij}(k) \mid \vartheta = m]$. Note that

$$E\left[\sum_{j\in\mathcal{N}_i(k)} \xi_{ij}(k)\middle|\vartheta = m\right] = m.$$

Since all $\xi_{ij}(k)$ are identically distributed,

$$E[\xi_{ij}(k) \mid \vartheta = m] = \frac{m}{|\mathcal{N}_i(t_k)|}.$$

This implies

$$E\left[\zeta_i(k)\right] = \int_{-\infty}^{\infty} \frac{1}{m} \sum_{j\in\mathcal{N}_i(k)} \frac{m}{|\mathcal{N}_i(t_k)|} x_j(k)f_\vartheta(m)\mathrm{d}m \tag{5.40}$$

$$= \frac{\sum\limits_{j\in\mathcal{N}_i(k)} x_j(k)}{|\mathcal{N}_i(t_k)|} \int_{-\infty}^{\infty} f_\vartheta(m)\mathrm{d}m \tag{5.41}$$

$$= \frac{\sum\limits_{j\in\mathcal{N}_i(k)} x_j(k)}{|\mathcal{N}_i(t_k)|}. \tag{5.42}$$

By this, the proof is concluded. □

By Proposition 25, protocol (5.6) behaves, in expectation, as the ideal protocol in which each agent obtains the linear average of neighbors' states, see, e.g., [13, 44], in case all incoming arcs weights are the same. Then, in expectation, system (5.10) exhibit the same iteration dynamics as a traditional protocol. This results in having, in expectation, the same number of iterations to consensus[14] as a traditional protocol. However, by Section 4, every iteration of the traditional protocol requires $\sum_{i=1}^{n} |\mathcal{N}_i(k)|$ orthogonal transmissions, whilst our approach requires only 2. Quantifying the expected amount of saved wireless resources is straightforward.

[14] Until, e.g., $\|\boldsymbol{x}(k) - \boldsymbol{x}^*\| < \epsilon$, for a given ϵ.

Worst case scenario.

Let $\epsilon \in \mathbb{R}$ such that, if

$$\|\boldsymbol{x}(\kappa) - \boldsymbol{x}^*\| < \epsilon, \tag{5.43}$$

we say that the system achieves the desired result at iteration $\kappa \in \mathbb{N}_0$. A system composed of n agents, with initial information states stacked in vector $\boldsymbol{x}(0)$, is given. It employs a traditional consensus algorithm[15] together with a standard communication system. It is assumed to achieve the desired result in κ_t iterations. By Section 4, to reach this end, it needs

$$\mathcal{I}_t = \sum_{t=1}^{\kappa_t} \sum_{i=1}^{n} |\mathcal{N}_i(t)|$$

orthogonal transmissions of signals, i.e., wireless resources. On the other hand, our consensus protocol achieves the result in $\kappa_b \in \mathbb{N}$ iterations, thus requiring, also by Section 4,

$$\mathcal{I}_b = 2\kappa_b$$

orthogonal transmissions of signals. Note that, in general, $\kappa_b \neq \kappa_t$, because of the impact of the fading channel that distorts the received signal by each agent. In fact, although the expected received signal equals the linear average of neighboring agents' states[16], the actual received signal is their convex combination, with normalized fading coefficients as parameters[17]. Our approach performs better (in terms of employed wireless resources) if $\mathcal{I}_b < \mathcal{I}_t$, equivalently, if

$$\kappa_b < \frac{1}{2} \sum_{t=1}^{\kappa_t} \sum_{i=1}^{n} |\mathcal{N}_i(t)|. \tag{5.44}$$

In case of a time-invariant underlying network topology, the latter inequality becomes

$$\kappa_b < \frac{\sum_{i=1}^{n} |\mathcal{N}_i|}{2} \kappa_t. \tag{5.45}$$

This shows that, as long as our protocol guarantees convergence to the desired result, i.e., (5.43), in at most $\frac{\sum_{i=1}^{n} |\mathcal{N}_i|}{2}$ times the iterations required by the traditional approach, wireless resources are saved. To properly address the distorting impact of the channel (affecting κ_b), in what follows, randomized simulations are performed.

[15] [13, Sec. II].

[16] See the previous paragraph.

[17] See Proposition 17.

Randomized analysis.

We analyze networks of $n \in \mathbb{N}$ agents, for n that varies from 3 to 50. Concerning the desired result, see (5.43), let

$$\epsilon = 0.1.$$

We initially compare our approach to traditional approaches for time-invariant underlying network topologies. For every cardinality of the nodes set, say $i \in [3, 50]$, we randomly generate 100 different strongly connected topologies. The initial information states are stacked in vector $\boldsymbol{x}(0)$, so that, $\forall i \in \mathcal{N}$, $[\boldsymbol{x}(0)]_i = i$. For each topology, two experiments are run[18] by employing the following approaches:

[18] In what follows, we assume $\sigma = 0.5$.

(i) **Traditional approach**: every agent has dynamics[19]

$$\forall k \in \mathbb{N}_0,\ x_i(k+1) = (1-\sigma)x_i(k) + \sigma \frac{\sum_{j \in \mathcal{N}_i(k)} x_j(k)}{|\mathcal{N}_i(k)|}, \quad (5.46)$$

and every iteration requires $\sum_{i=1}^{n} |\mathcal{N}_i(k)|$ orthogonal channel accesses (wireless resources). Iterations to consensus (see (5.43)) are denoted by κ_t. By $\mathcal{I}_t$ we denote the wireless resources (orthogonal transmissions) employed to reach κ_t iterations.

[19] Similarly to the traditional approach already revised in (2.12).

(ii) **Interference-based approach**: every agent has dynamics (5.10) and each iteration update requires 2 orthogonal accesses of the channel. Iterations to consensus (see (5.43)) are denoted by κ_b. By $\mathcal{I}_b$ we denote the wireless resources (orthogonal transmissions) employed to reach κ_b iterations.

Channel coefficients are independent (see the paragraph about comparison in expectation) and drawn out of an uniform distribution. Wireless resources required by the two different approaches are compared, such that, for

$$\frac{\mathcal{I}_t}{\mathcal{I}_b} > 1,$$

equivalently ,

$$\log_{10}\left(\frac{\mathcal{I}_t}{\mathcal{I}_b}\right) > 0,$$

the broadcast approach requires less wireless resources than the traditional approach to reach the desired result.

Figure 5.19 illustrates (orange) the comparison, between the traditional approach and our approach, of iterations required to converge, i.e., $\frac{\kappa_t}{\kappa_b}$. The traditional approach requires almost always less iterations to converge to consensus than ours. However, the comparison of employed wireless resources (blue), i.e., $\frac{\mathcal{I}_t}{\mathcal{I}_b}$, proves that our approach performs much better, especially for networks with more than 10 nodes.

The same comparison is done for consensus problems involving sequences of time-varying network topologies. We limit our analysis to the case of sequences of strongly connected graphs, see Proposition 22. For every nodes set's cardinality, we randomly generate a sequence[20]

[20] Ideally infinite.

of strongly connected graphs. For every sequence, we repeat the experiment done for the time-invariant case.

In Figure 5.20, the same general analysis done for Figure 5.19 can be employed. The traditional approach guarantees much less iterations until agreement (orange), whereas the wireless resources that our approach requires (blue) are much less than the ones traditionally required.

The randomized analysis of the employed wireless resources reinforces what already shown for the comparison in expectation and worst case. Although our approach requires in general more iterations until consensus, mostly due to the presence of the unknown channel, it requires fewer wireless resources overall. The savings are higher for network with more nodes.

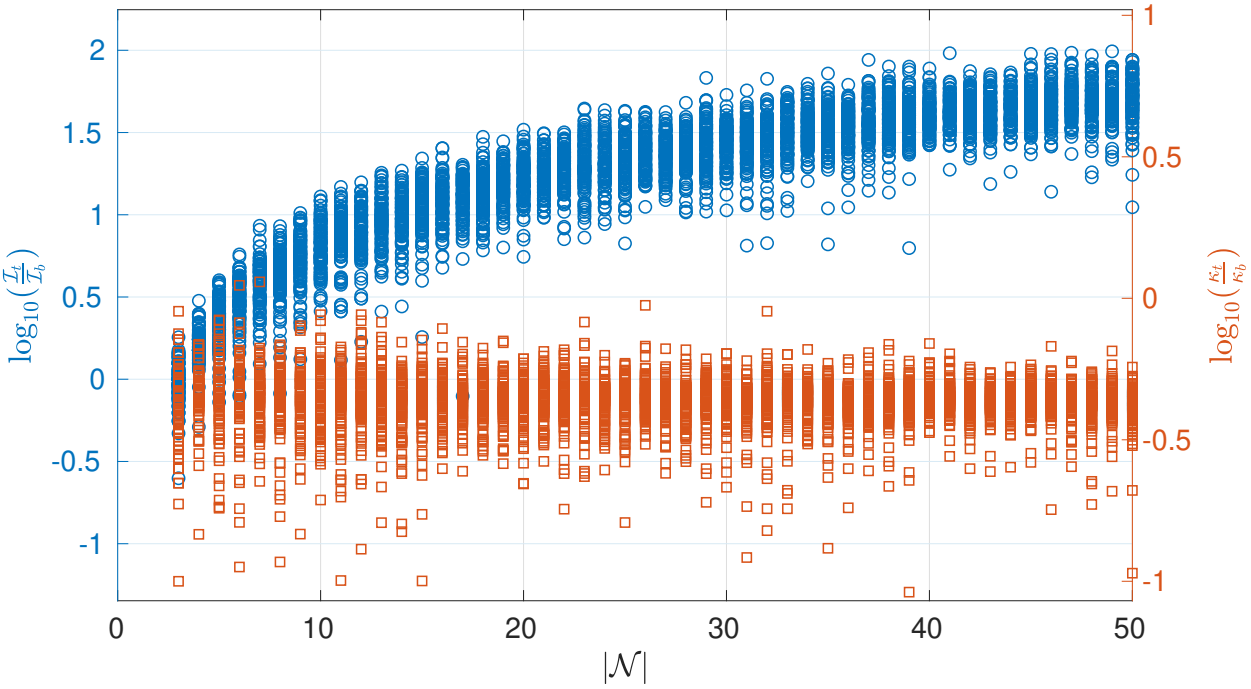


Figure 5.19: Comparison of broadcast and traditional approaches for time-invariant systems.

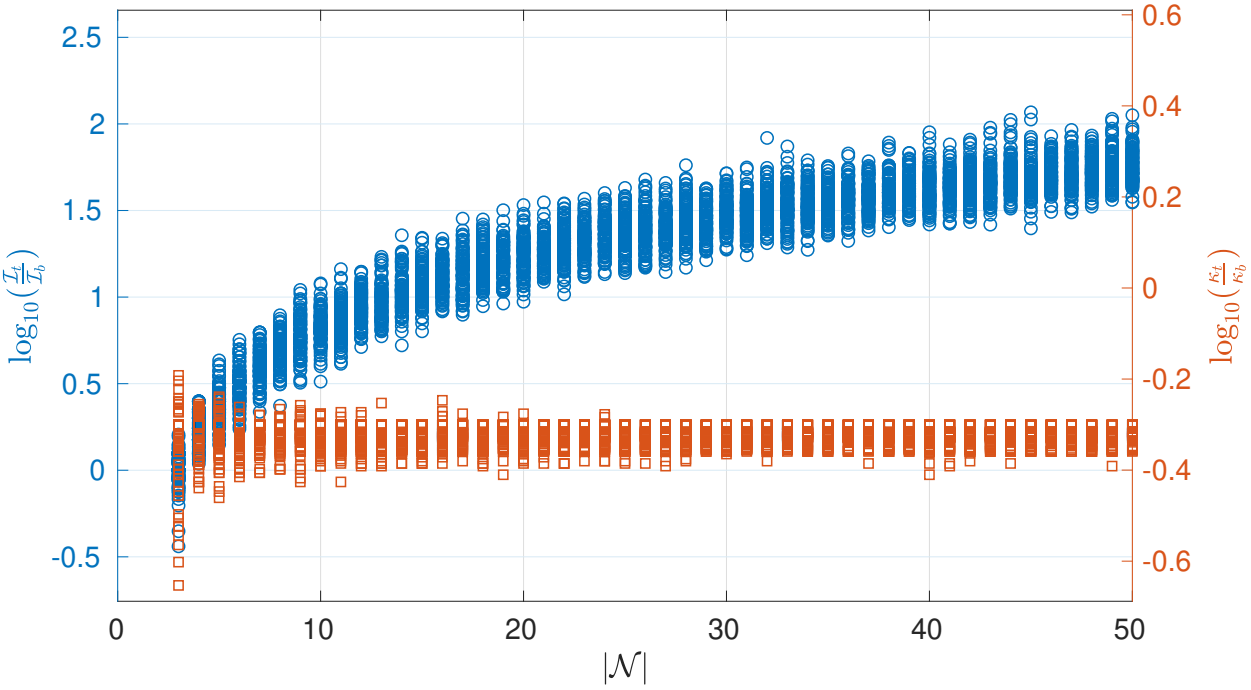


Figure 5.20: Comparison of broadcast and traditional approaches for time-invariant systems.

5.2 *Exploiting Wireless Interference For Distributively Solving Linear Equations*

Partial results of the presented section are in Molinari et al. [61]. The candidate is the first author of the contribution.

Beginning from [71], literature has exhibited an increasing interest towards networks of cooperative autonomous agents solving linear algebraic equations, i.e.,

$$A\mathbf{x} = b,$$

in a distributed way, see, e.g., [72], [101], and [73]. Each agent can access only a distinct partition of the equation (i.e., a subset of rows of A and b) and aims at retrieving the global solution by cooperating with other agents. [71] considers a simple setting in which A is a nonsingular square matrix and the underlying network topology is fixed; their approach guarantees exponentially fast convergence to the global solution. Later on, many contributions extended this first result, e.g., [51] considers non-square A, nonunique solutions, jointly-connected time-varying network topologies, and asynchronous communication. As shown in [73], this result involves many possible applications, e.g., solving least square or *network localization* problems. Also, by [3], the cooperative solution of linear equations finds an important application in clustered computation involving sensitive data, such as business financial records, personally identifiable health information, etc.

For distributively solving linear algebraic equations, most literature has been focusing on improving the convergence rate and on relaxing conditions on neighbor graphs, see [51, Table I]. To the best of our knowledge, the impact of the communication medium has not been taken into account. Indeed, in the context of this thesis, the current section intends to study how interference can be exploited for distributively solving a linear algebraic equation. This approach is argued to have additional benefits:

- **Privacy**. Sets of neighbors and arc weights are unknown to agents, are extensively motivated in the previous sections. Therefore, it is impossible to use the received signals to have access to neighboring agents' local equations. This is a useful feature when different agents are not in the same domain of trust and each local equation may contain sensitive information, see, e.g., [3] and [110].
- **Saving resources**. Exploiting interference instead of getting rid of it allows for saving wireless resources, see Section 5.1.4.

5.2.1 *Problem Description*

A group of $\nu \in \mathbb{N}$ autonomous agents, grouped in the set $\mathcal{N} := \{1 \dots \nu\}$, need to cooperate to solve the linear algebraic equation

$$A\mathbf{x} = b, \tag{5.47}$$

where $A \in \mathbb{R}^{n \times m}$, $b \in \mathbb{R}^n$, $\mathbf{x} \in \mathbb{R}^m$, and $\nu \leq n \leq m$. Each agent $i \in \mathcal{N}$ has access only to a distinct[21] subset of $n_i \in \mathbb{N}$ rows of (5.47), i.e., it

[21] By distinct, we mean "free of overlap".

can locally solve

$$A_i \boldsymbol{x}_i = b_i, \tag{5.48}$$

where $A_i \in \mathbb{R}^{n_i \times m}$, $b_i \in \mathbb{R}^{n_i}$, and $\boldsymbol{x}_i \in \mathbb{R}^m$. Clearly, $\sum_{i \in \mathcal{N}} n_i = n$. Agents communicate over the wireless channel and exploit its interference property as proposed in Section 5.1.1. In Molinari et al. [61], we consider $\mathfrak{G}$ to be a sequence of fully-connected graphs. Throughout this section, the aim is to relax this condition and introduce the concept of *repeatedly jointly $\mathcal{D}$-connectedness*, reviewing some results contained in [51].

Definition 16. *Any nonempty subset $\mathcal{E} \subseteq \mathcal{N}$ is said to be* fully populated *if*

$$\bigcap_{i \in \mathcal{E}} \ker(A_i) = \bigcap_{i \in \mathcal{N}} \ker(A_i).$$

Note that, in case of a unique solution, if $\mathcal{E}$ is fully populated, then $\cap_{i \in \mathcal{E}} \ker(A_i) = \{0\}$.

Remark 8. *Since each agent has access only to a distinct subset or rows, if $\mathcal{E} \subset \mathcal{N}$ is fully populated, it implies that A has some linearly dependent rows.*

Definition 17. *Any subset which is not fully populated is* partially populated.

Let $2^{\mathcal{N}}$ denote the power set of $\mathcal{N}$, namely, the set of all subsets of $\mathcal{N}$, including the empty set and $\mathcal{N}$ itself.

Definition 18. *A collection of nonempty proper subsets $\mathcal{W} \subset 2^{\mathcal{N}}$ is connected by the graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ if each subset contains at least one $i \in \mathcal{N}$ with a neighbor in $\mathcal{N} \setminus \mathcal{E}$.*

Definition 19. *Let $\mathcal{D} \subset 2^{\mathcal{N}}$ be the collection of all partially populated subsets of $\mathcal{N}$. A graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ is* $\mathcal{D}$-connected *if $\mathcal{D}$ is connected by $\mathcal{G}$.*

This latter definition states that a time-invariant graph is $\mathcal{D}$-connected if each partially populated subset of $\mathcal{N}$ has at least one neighbor in the rest of the graph. It is clear that the concept of $\mathcal{D}$-connectivity depends on the matrix A. The same graph can be $\mathcal{D}$-connected (or not) depending on the specific A. The following example clarifies the concepts.

Example 16. *Let*

$$A := \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 0 \\ 2 & 4 & 6 \end{bmatrix},$$

$\nu = 3$, and, $\forall i \in \mathcal{N}$, $n_i = 1$. It is clear that $\text{rank}(A) = 2 = m - 1$, *since the first and the third rows are linearly dependent. Thus, $\mathcal{D} = \{\{1\}, \{2\}, \{3\}, \{1, 3\}\}$.*
Figure 5.21 shows a $\mathcal{D}$-connected graph for matrix A. Note that the same graph is not strongly connected.

Focusing on sequences of time-varying graphs, we can define the following items.

Definition 20. *A finite sequence of ℓ directed graphs $\mathfrak{G}$ is jointly $\mathcal{D}$-connected if the composition $\mathcal{G}(\ell) \circ \mathcal{G}(\ell - 1) \circ \cdots \circ \mathcal{G}(1)$ is $\mathcal{D}$-connected.*

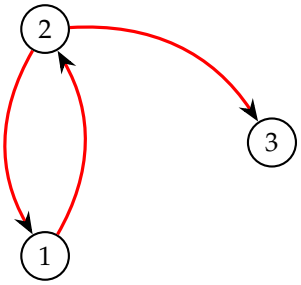


Figure 5.21: $\mathcal{D}$-connected graph for matrix A of Example 16.

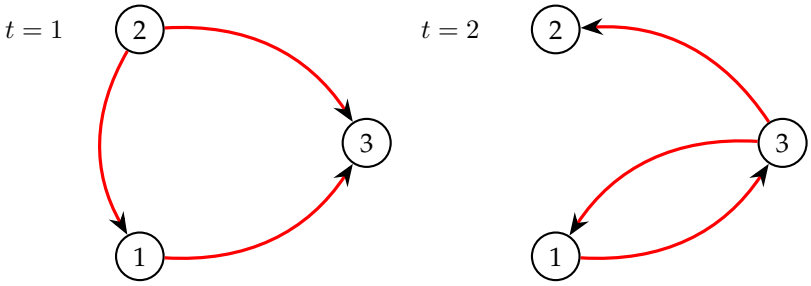


Figure 5.22: Jointly $\mathcal{D}$-connected graph sequence for matrix A of Example 16.

Example 17. *Consider matrix A of Example 16. The sequence of $\ell = 2$ graphs in Figure 5.22 is jointly $\mathcal{D}$-connected.*

Definition 21. *An infinite sequence of directed graphs $\mathfrak{G}$ is* repeatedly jointly $\mathcal{D}$-connected by sub-sequences of length ℓ *if it exists a positive integer ℓ, such that, $\forall k \in \mathbb{N}_0$, the sequence $\mathcal{G}(kl), \ldots, \mathcal{G}((k+1)l - 1)$ is jointly $\mathcal{D}$-connected.*

5.2.2 Communication Protocol

As in Section 5.1.1, each agent $j \in \mathcal{N}$ at every iteration $k \in \mathbb{N}_0$ broadcasts two pieces of information, i.e., the m-dimensional vector

$$\boldsymbol{\tau}_j^a(k) := \boldsymbol{x}_j(k)$$

and

$$\tau_j^b(k) := 1.$$

Note that these two signals are broadcast orthogonally, i.e., independently from each other (e.g., on different frequencies). For the synchronous broadcast of these two signals, at every algorithm iteration, agents need $m + 1$ orthogonal transmissions (namely, $m + 1$ wireless resources are used per every iteration), since $\boldsymbol{\tau}_j^a(k) \in \mathbb{R}^m$ and $\tau_j^b(k) \in \mathbb{R}$. According to the WMAC model, at every iteration $k \in \mathbb{N}_0$ each agent $i \in \mathcal{N}$ receives

$$\boldsymbol{\nu}_i^a(t_k) := \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) \boldsymbol{\tau}_j^a(k) = \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) \boldsymbol{x}_j(k) \tag{5.49}$$

and

$$\nu_i^b(t_k) := \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) \tau_j^b(k) = \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k). \tag{5.50}$$

Fading coefficients have already been discussed in Section 4. Among the benefits of exploiting the interference, we acknowledged privacy. In fact, being fading coefficients unknown to agents, it is utterly impossible for any agent $i \in \mathcal{N}$ to reconstruct neighbors' local information by knowing $\boldsymbol{\nu}_i^a(t_k)$ and $\nu_i^b(t_k)$.

5.2.3 *Algorithm*

Each agent $i \in \mathcal{N}$, at every iteration $k \in \mathbb{N}_0$, obtains $\boldsymbol{\nu}_i^a(t_k)$ and $\nu_i^b(t_k)$, which may be used for updating $\boldsymbol{x}_i(k)$. $\boldsymbol{\nu}_i^a(t_k)$ represents a linear combination of the local estimates $\boldsymbol{x}_j(k)$ of all agents that can transmit information to agent i. $\nu_i^b(t_k)$ is the sum of the corresponding channel coefficients. The proposed algorithm is executed $\forall i \in \mathcal{N}$ and $\forall k \in \mathbb{N}_0$ and is as follows:

$$\boldsymbol{x}_i(k+1) = \boldsymbol{x}_i(k) - \frac{1}{\nu_i^b(t_k)} P_i \left(\nu_i^b(t_k) \boldsymbol{x}_i(k) - \boldsymbol{\nu}_i^a(t_k) \right), \tag{5.51}$$

where $P_i \in \mathbb{R}^{m \times m}$ is the orthogonal projection matrix onto the kernel of A_i. $\forall i \in \mathcal{N}$, the orthogonal projection matrix P_i is computed as follows, see [102, Ch. 7.3]:

$$P_i := \kappa_i (\kappa_i' \kappa_i)^{-1} \kappa_i',$$

where $\kappa_i \in \mathbb{R}^{m \times \rho}$, $\rho = \text{nullity}(A_i)$, and its columns form a basis for the kernel of A_i.

Remark 9. $\forall \boldsymbol{v} \in \mathbb{R}^m$, $P_i \boldsymbol{v} \in \ker(A)_i$.

Remark 10. $\forall \boldsymbol{v} \in \mathbb{R}^m$, $A_i P_i \boldsymbol{v} = \mathbf{0}_{n_i}$.

Proposition 26. *If $\boldsymbol{x}_i(k)$ solves (5.48) and $\boldsymbol{x}_i(k)$ is updated according to (5.51), then $\boldsymbol{x}_i(k+1)$ solves (5.48).*

Proof. $\boldsymbol{x}_i(k+1)$ solves (5.48) iff $A_i \boldsymbol{x}_i(k+1) = b_i$, i.e., by (5.51),

$$A_i \left(\boldsymbol{x}_i(k) - \frac{1}{\nu_i^b(t_k)} P_i \left(\nu_i^b(t_k) \boldsymbol{x}_i(k) - \boldsymbol{\nu}_i^a(t_k) \right) \right) = b_i.$$

By Remark 10, the latter becomes

$$A_i \boldsymbol{x}_i(k) - \frac{1}{\nu_i^b(t_k)} A_i P_i \left(\nu_i^b(t_k) \boldsymbol{x}_i(k) - \boldsymbol{\nu}_i^a(t_k) \right) = A_i \boldsymbol{x}_i(k),$$

which is, by hypothesis of the proposition, equal to b_i. This concludes the proof. □

Convergence to a global solution can be formalized as follows.

Theorem 8 (Unique solution)**.** *A set $\mathcal{N}$ of communicating agents update their estimates according to (5.51). If $\mathfrak{G}$ is a sequence of graphs repeatedly jointly $\mathcal{D}$-connected by sub-sequences of length ν, then all $\boldsymbol{x}_i(k)$ converge to the global solution of (5.47) notwithstanding the unknown fading channel.*

5.2.4 *Proof of Theorem 8*

The proof of Theorem 8 is inspired by [51] and extends the proofs of [71] to the case of communication interference. We first deal with the case of (5.47) having a unique solution (i.e. $m = n$); after that, we extend the proof to the more general case (i.e. $m \geq n$).

5.2.4.1 *Unique solution*

Let $x^* \in \mathbb{R}^m$ be the unique solution of (5.47). Let's define error variables, i.e.,

$$\forall i \in \mathcal{N},\ \forall k \in \mathbb{N}_0, \quad \boldsymbol{e}_i(k) := \boldsymbol{x}_i(k) - x^*. \tag{5.52}$$

Clearly, all agents have the global solution if and only if, $\forall i \in \mathcal{N},\ \boldsymbol{e}_i(k) = \mathbf{0}_m$. Inserting (5.8) into (5.51) yields, $\forall k \in \mathbb{N}_0,\ \forall i \in \mathcal{N}$,

$$\boldsymbol{x}_i(k+1) = \boldsymbol{x}_i(k) - P_i \left(\boldsymbol{x}_i(k) - \sum_{j=1}^{\nu} h_{ij}(k) \boldsymbol{x}_j(k) \right). \tag{5.53}$$

Note that also the individual normalized coefficients $h_{ij}(k)$ in (5.53) are unknown. The only available information is that they sum up to 1. In order to bring (5.53) to a compact form, let $\boldsymbol{x}(k) \in \mathbb{R}^{\nu m}$ be the column vector stacking local solutions, i.e.,

$$\forall k \in \mathbb{N}_0, \quad \boldsymbol{x}(k) := [\boldsymbol{x}_1(k)', \dots, \boldsymbol{x}_\nu(k)']'.$$

Similarly, let $\boldsymbol{x}^* \in \mathbb{R}^{\nu m}$ be the column stacking ν vectors x^*, i.e. $\boldsymbol{x}^* = [x^{*'}, \dots, x^{*'}]'$. Moreover, let $P \in \mathbb{R}^{\nu m \times \nu m}$ be the blockdiagonal matrix composed of the ν orthogonal projection matrices P_i, i.e.,

$$P := \text{blockdiag}\left(P_1, \dots, P_\nu\right).$$

Define the matrix $H(k) \in \mathbb{R}^{\nu \times \nu}$,

$$\forall k \in \mathbb{N}_0,\ \forall i \in \mathcal{N},\ \forall j \in \mathcal{N}, \quad [H(k)]_{ij} = h_{ij}(k).$$

Clearly, $H(k)$ is row-stochastic, $\forall k \in \mathbb{N}_0$. Iterate (5.53) can now be rewritten as, $\forall k \in \mathbb{N}_0$,

$$\boldsymbol{x}(k+1) = \boldsymbol{x}(k) - P\left(\boldsymbol{x}(k) - (H(k) \otimes \mathbb{I}_m)\boldsymbol{x}(k)\right), \tag{5.54}$$

where $H(k) \otimes \mathbb{I}_m$ denotes the Kronecker product of $H(k)$ and $\mathbb{I}_m$. Let, $\forall k \in \mathbb{N}_0$, $\boldsymbol{e}(k) \in \mathbb{R}^{\nu m}$ stack all $\boldsymbol{e}_i(k)$, i.e.,

$$\boldsymbol{e}(k) := [\boldsymbol{e}_1(k)', \dots, \boldsymbol{e}_\nu(k)']'. \tag{5.55}$$

Lemma 5. $\forall k \in \mathbb{N}_0$, $(H(k) \otimes \mathbb{I}_m)\boldsymbol{x}^* = \boldsymbol{x}^*$.

Proof.

$$(H(k) \otimes \mathbb{I}_m)\boldsymbol{x}^* = \begin{pmatrix} \sum_{j=1}^{\nu} h_{1j}(k) x^* \\ \vdots \\ \sum_{j=1}^{\nu} h_{\nu j}(k) x^* \end{pmatrix} = \boldsymbol{x}^*,$$

where the second equality is a consequence of Observation 1 on pg. 65. □

Lemma 6. $\forall k \in \mathbb{N}_0$, $P\boldsymbol{e}(k) = \boldsymbol{e}(k)$.

Proof. Since $\boldsymbol{x}_i(k)$ is a local solution of (5.47), $A_i\boldsymbol{e}_i(k) = A_i(\boldsymbol{x}_i(k) - x^*) = b_i - b_i = 0$. Therefore, $\boldsymbol{e}_i(k) \in \ker(A_i)$. $\forall i \in \mathcal{N}$, P_i is the orthogonal projection onto the kernel of A_i, therefore, since $\boldsymbol{e}_i(k)$ is already in $\ker(A_i)$,

$$P_i\boldsymbol{e}_i(k) = \boldsymbol{e}_i(k).$$

This is valid $\forall i \in \mathcal{N}$, thus the proof is concluded. □

By Lemma 5 and by subtracting $\boldsymbol{x}^*$ from both sides of (5.54), one gets

$$\begin{aligned}\boldsymbol{e}(k+1) &= \boldsymbol{e}(k) - P\left(\boldsymbol{x}(k) - (H(k) \otimes \mathbb{I}_m)\boldsymbol{x}(k)\right) \\ &= \boldsymbol{e}(k) - P\left(\boldsymbol{x}(k) - (H(k) \otimes \mathbb{I}_m)\boldsymbol{x}(k) + (H(k) \otimes \mathbb{I}_m)\boldsymbol{x}^* - \boldsymbol{x}^*\right) \\ &= \boldsymbol{e}(k) - P\left(\boldsymbol{e}(k) - (H(k) \otimes \mathbb{I}_m)\boldsymbol{e}(k)\right)\end{aligned}$$

By Lemma 6, the latter becomes, $\forall k \in \mathbb{N}_0$,

$$\boldsymbol{e}(k+1) = Q(k)\boldsymbol{e}(k), \tag{5.56}$$

where $Q(k) := P\left(H(k) \otimes \mathbb{I}_m\right)$. If the matrix sequence $\{Q(k)\}_{k\in\mathbb{N}_0}$ converges exponentially, Theorem 8 is proven. Similarly to [51], we are going to employ a *mixed norm* for proving this.

Next, we discuss the mixed l_2/l_∞ vector norm and its induced matrix norm. In literature, the concept of mixed matrix norm is closely related to the subject of *norm compression of block-partitioned matrices*, see, e.g., [7], and it has been employed in the context of *compressed sensing*, see [28].

Assume $\boldsymbol{v} \in \mathbb{R}^{dm}$, $d, m \in \mathbb{N}$. Partition $\boldsymbol{v}$ as $\boldsymbol{v} = [\tilde{\boldsymbol{v}}_1', \ldots, \tilde{\boldsymbol{v}}_m']'$, where $[\tilde{\boldsymbol{v}}_i]_j := [\boldsymbol{v}]_{(i-1)d+j}$, $i = 1, \ldots, m$, $j = 1, \ldots, d$. Define $\boldsymbol{w} \in \mathbb{R}^m_{\geq 0}$ by $[\boldsymbol{w}]_i = \|\tilde{\boldsymbol{v}}_i\|_2$, $i = 1, \ldots, m$. Then the mixed l_2/l_∞ norm of $\boldsymbol{v}$ corresponding to the integer d is defined as

$$\|\boldsymbol{v}\|_{2,\infty} := \|\boldsymbol{w}\|_\infty . \tag{5.57}$$

Assume $A \in \mathbb{R}^{dm\times dm}$, $d, m \in \mathbb{N}$. Partition A as

$$A = \begin{bmatrix} \tilde{A}_{11} & \ldots & \tilde{A}_{1m} \\ \ldots & \ldots & \ldots \\ \tilde{A}_{m1} & \ldots & \tilde{A}_{mm} \end{bmatrix},$$

where $\tilde{A}_{ij} \in \mathbb{R}^{d\times d}$ is given by $[\tilde{A}_{ij}]_{kl} := [A]_{(i-1)d+k,(j-1)d+l}$, $i, j = 1, \ldots, m$ and $k, l = 1, \ldots, d$. Define $B \in \mathbb{R}^{m\times m}_{\geq 0}$ by $[B]_{ij} = \left\|\tilde{A}_{ij}\right\|_2$ where the latter is the matrix norm induced by the l_2 vector norm. Then the mixed l_2/l_∞ norm of matrix A corresponding to the integer d is defined as

$$|||A|||_{2,\infty} := \|B\|_\infty ,$$

where $\|B\|_\infty$ is the norm of matrix B induced by the l_∞ vector norm. It is straightforward to show that $\|\cdot\|_{2,\infty}$ and $|||\cdot|||_{2,\infty}$ indeed satisfy all

norm axioms. In Appendix A.3.2, we show that $|||\cdot|||_{2,\infty}$ is the matrix norm induced by the vector norm $\|\cdot\|_{2,\infty}$, i.e.

$$|||A|||_{2,\infty} = \sup_{\boldsymbol{v}\neq 0} \frac{\|A\boldsymbol{v}\|_{2,\infty}}{\|\boldsymbol{v}\|_{2,\infty}}.$$

Furthermore, [51, Lemma 3] have shown that $|||\cdot|||_{2,\infty}$ is submultiplicative, i.e., $|||A_1A_2|||_{2,\infty} \leq |||A_1|||_{2,\infty}\,|||A_2|||_{2,\infty}$.

Lemma 7 ([72, Lemma 2]). *For any non-empty set of $m \times m$ real orthogonal projection matrices $\{T_1, \ldots, T_\ell\}$,*

$$\|T_\ell T_{\ell-1} \cdot \cdots \cdot T_1\|_2 \leq 1. \tag{5.58}$$

In particular,

$$\|T_\ell T_{\ell-1} \cdot \cdots \cdot T_1\|_2 < 1 \tag{5.59}$$

if and only if

$$\dim\left(\bigcap_{i=1}^{\ell} \text{image}(T_i)\right) = 0. \tag{5.60}$$

Proof. See [72]. □

Corollary 5. *Given the projection matrices employed in (5.51),*

$$\|P_{\ell_1} \cdot \cdots \cdot P_{\ell_\nu}\|_2 < 1, \tag{5.61}$$

for $\{\ell_1 \ldots \ell_\nu\} = \{1 \ldots \nu\}$.

Proof. In case (5.47) has a unique solution, $\dim(\ker(A)) = 0$. The latter implies that $\dim(\bigcap_{i=1}^{\nu} \ker(A_i)) = 0$. Since, $\forall i \in \mathcal{N}$, P_i is the orthogonal projection matrix onto the kernel of A_i, $\ker(A_i) = \text{image}(P_i)$, then

$$\dim\left(\bigcap_{i=1}^{\nu} \text{image}(P_i)\right) = 0,$$

which, by Lemma 7, implies (5.61). This concludes the proof. □

At this point, the concept of *repeatedly jointly $\mathcal{D}$-connectedness of length ν* comes into play.

Corollary 6. *If $\mathfrak{G}$ is a sequence of graphs repeatedly jointly $\mathcal{D}$-connected by sub-sequences of length ν, for each iteration index $k \in \mathbb{N}_0$, there exists a set $\{\ell_1 \ldots \ell_\nu\} = \{1 \ldots \nu\}$ such that*

$$\|P_{\ell_1} \cdot \cdots \cdot P_{\ell_\nu}\|_2 < 1 \tag{5.62}$$

and

$$h_{\ell_1\ell_2}(k+1) \cdot \cdots \cdot h_{\ell_{\nu-1}\ell_\nu}(k+\nu-1) > 0. \tag{5.63}$$

Proof. Inequality (5.62) follows directly from Corollary 5.

Since (5.47) has a unique solution, by Definition 16, the only fully populated subset of $\mathcal{N}$ is $\mathcal{N}$ itself. This implies that set $\mathcal{D} = 2^{\mathcal{N}} \setminus \{\emptyset, \mathcal{N}\}$ determines $\mathcal{D}$-connectedness. By the latter and Definition 21, $\forall k \in \mathbb{N}_0$, there exists a sequence of arcs

$$\{(\ell_h, \ell_{h+1}) \in \mathcal{A}(k+h)\}_{h=1,\ldots,\nu-1},$$

with, $\forall h = \{1, \dots, \nu\}$, $\ell_h \in \mathcal{N}$ and $\ell_{h_a} \neq \ell_{h_b}$, for $h_a \neq h_b$. By this latter consideration and by definition of normalized channel coefficients,

$$\forall h = \{1, \dots, \nu - 1\},\ h_{\ell_h \ell_{h+1}}(k+h) > 0,$$

thus yielding (5.63). The proof is concluded. □

Lemma 8. $\exists \gamma \in (0,1)$*:*

$$\forall k \in \mathbb{N}_0, \quad \|\boldsymbol{e}(k+\nu)\|_{2,\infty} \leq \gamma \|\boldsymbol{e}(k)\|_{2,\infty}. \tag{5.64}$$

Proof. By (5.56),

$$\begin{aligned}\boldsymbol{e}(k+\nu) &= Q(k+\nu-1) \cdot \dots \cdot Q(k)\boldsymbol{e}(k),\\ &= \underbrace{P\left(H(k+\nu-1) \otimes \mathbb{I}_m\right) \cdot \dots \cdot P\left(H(k) \otimes \mathbb{I}_m\right)}_{:=Q_k^\nu} \boldsymbol{e}(k).\end{aligned}$$

Matrix $Q_k^\nu \in \mathbb{R}^{\nu m \times \nu m}$ can be seen as a block-matrix composed of $\nu \times \nu$ blocks of dimension $m \times m$. Each block i, j ($i, j \in \{1 \dots \nu\}$) is denoted by $Q_k^\nu[i,j]$ and is explicitly written as

$$\begin{aligned}Q_k^\nu[i,j] = \sum_{\ell_1=1}^{m} \cdots \sum_{\ell_{\nu-1}=1}^{m} & P_i h_{i\ell_1}(k) P_{\ell_1} h_{\ell_1\ell_2}(k+1) \cdot \ \dots \\ & \dots \ \cdot P_{\ell_{\nu-2}} h_{\ell_{\nu-2}\ell_{\nu-1}}(k+\nu-2) P_{\ell_{\nu-1}} h_{\ell_{\nu-1}j}(k+\nu-1).\end{aligned}$$

By the triangle inequality for norms,

$$\begin{aligned}\|Q_k^\nu[i,j]\|_2 \leq \sum_{\ell_1=1}^{m} \cdots \sum_{\ell_{\nu-1}=1}^{m} & \left\|P_i \cdot \dots \cdot P_{\ell_{\nu-1}}\right\|_2 \cdot \\ & \cdot h_{i\ell_1}(k) \cdot \dots \cdot h_{\ell_{\nu-1}j}(k+\nu-1).\end{aligned} \tag{5.65}$$

In general, by Lemma 7,

$$\forall \{i, \ell_1, \dots, \ell_{\nu-1}\},\ \left\|P_i P_{\ell_1} \cdot \dots \cdot P_{\ell_{\nu-1}}\right\|_2 \leq 1. \tag{5.66}$$

Among all considered ν-dimensional sets $\{i, \ell_1, \dots, \ell_{\nu-1}\}$ in (5.66), there is also set $\{1 \dots \nu\}$. For this set, by Corollary 5,

$$\left\|P_i P_{\ell_1} \cdot \dots \cdot P_{\ell_{\nu-1}}\right\|_2 < 1. \tag{5.67}$$

and

$$h_{i\ell_1}(k) h_{\ell_1\ell_2}(k+1) \cdot \dots \cdot h_{\ell_{\nu-1}j}(k+\nu-1) > 0. \tag{5.68}$$

Inserting (5.67) and (5.68) into (5.65) yields

$$\|Q_k^\nu[i,j]\|_2 < \sum_{\ell_1=1}^{m} \cdots \sum_{\ell_{\nu-1}=1}^{m} h_{i\ell_1}(k) \cdot \dots \cdot h_{\ell_{\nu-1}j}(k+\nu-1). \tag{5.69}$$

In fact, one of the summands in (5.65), i.e., the summand corresponding to (5.67)-(5.68), is strictly smaller than $h_{i\ell_1}(k) h_{\ell_1\ell_2}(k+1) \cdot \dots \cdot h_{\ell_{\nu-1}j}(k+\nu-1)$.

To prove the Lemma, we need to show that

$$\forall k \in \mathbb{N}_0,\ \left\|\left|Q_k^\nu\right|\right\|_{2,\infty} \leq \gamma < 1.$$

By definition of l_2/l_∞ mixed matrix norm,

$$\left|\left|\left|Q_k^\nu\right|\right|\right|_{2,\infty} = \max_{i=1\dots m}\left(\sum_{j=1}^{m}\left\|Q_k^\nu[i,j]\right\|_2\right). \tag{5.70}$$

By Observation 1 on page 65, the rows of $H(k)$ sum up to 1, $\forall k \in \mathbb{N}_0$, therefore, $\forall k \in \mathbb{N}_0$, $\forall i \in \mathcal{N}$,

$$\sum_{j=1}^{m}\sum_{\ell_1=1}^{m}\cdots\sum_{\ell_{\nu-1}=1}^{m} h_{i\ell_1}(k)\cdot\dots\cdot h_{\ell_{\nu-1}j}(k+\nu-1) =$$

$$= \left(\sum_{\ell_1=1}^{m} h_{i\ell_1}(k)\cdot\dots\cdot\left(\sum_{j=1}^{m} h_{\ell_{\nu-1}j}(k+\nu-1)\right)\cdots\right) = 1.$$

Hence, using (5.69),

$$\forall k \in \mathbb{N}_0,\ \forall i \in \mathcal{N}, \quad \sum_{j=1}^{m}\left\|Q_k^\nu[i,j]\right\|_2 < 1,$$

thus, by (5.70),

$$\forall k \in \mathbb{N}_0, \quad \left|\left|\left|Q_k^\nu\right|\right|\right|_{2,\infty} < 1.$$

Consequently,[22]

$$\gamma := \sup_{k\in\mathbb{N}_0}\left|\left|\left|Q_k^\nu\right|\right|\right|_{2,\infty} < 1. \tag{5.71}$$

[22] The following condition holds in case the set of all possible Q_k^ν is a finite-set. This is the case if data-quantization is taken into account. See Assumption 5 for a longer explanation.

By definition of the induced vector norm,

$$\forall k \in \mathbb{N}_0, \quad \left\|\boldsymbol{e}(k+\nu)\right\|_{2,\infty} \le \left|\left|\left|Q_k^\nu\right|\right|\right|_{2,\infty}\left\|\boldsymbol{e}(k)\right\|_{2,\infty},$$

which, by (5.71), yields (5.64) thus concluding the proof of Lemma 8. □

The proof for Theorem 8 follows right from Lemma 8. In fact, by (5.64),

$$\lim_{k\to\infty}\left\|\boldsymbol{e}(k\nu)\right\|_{2,\infty} \le \gamma^k\left\|\boldsymbol{e}(0)\right\|_{2,\infty}. \tag{5.72}$$

Since $\|\cdot\|_{2,\infty}$ is a norm (5.72) yields that the error approaches 0 exponentially, therefore the local estimates $\boldsymbol{x}_i(k)$ approach x^* exponentially (independently of the unknown and time-varying normalized channel coefficients $h_{ij}(k)$).

5.2.4.2 *Multiple solutions*

Theorem 9 (Multiple solutions)**.** *A set $\mathcal{N}$ of communicating agents update their estimates according to (5.51). If $\mathfrak{G}$ is a sequence of graphs repeatedly jointly strongly[23] and $\mathcal{D}$-connected by sub-sequences of length ν, then all $\boldsymbol{x}_i(k)$ converge to the same global solution of (5.47) notwithstanding the unknown fading channel.*

[23] In [51], the necessary and sufficient condition for convergence is the graph sequence to be repeatedly jointly rooted and $\mathcal{D}$-connected. However, as explained in Chapter 2, throughout this work, we do not consider rooted graphs since, although they constitute a common necessary condition for consensus, they degenerate the concept of distributed control problem.

For proving convergence in case of multiple solutions, we make use of the same tools employed by [51, Sec. 4.2]. This is possible because, in the previous section, we have reduced (5.51) to a form similar to what presented in [51]. For this scenario, in fact, Corollary 6 cannot be used since having more than one solution implies that $\dim\left(\bigcap_{i=1}^{\nu}\ker(A_i)\right) \neq 0$.

Define the subspace $\mathcal{P}$ as

$$\mathcal{P} := \bigcap_{i=1}^{\nu} \text{image}(P_i) = \bigcap_{i=1}^{\nu} \ker(A_i),$$

and $\tilde{m} = m - \dim(\mathcal{P})$.

Definition 22. *Let the columns of the $m \times \tilde{m}$ matrix L' be an orthonormal basis for the orthogonal complement of $\mathcal{P}$.*

We define the following $\tilde{m} \times \tilde{m}$ matrix, $\forall i \in \mathcal{N}$,

$$\bar{P}_i := LP_iL'.$$

Lemma 9 ([72, Lemma 1])**.** *The following statements are true*

1. $\forall i \in \mathcal{N}$, $\bar{P}_i$ *is an orthogonal projection matrix;*
2. $\forall i \in \mathcal{N}$, $LP_i = \bar{P}_iL$
3. $\forall i \in \mathcal{N}$, $P_iL' = L'\bar{P}_i$*;*
4. $\bigcap_{i=1}^{\nu} \text{image}(\bar{P}_i) = 0$.

Proof. See [72, Lemma 1]. □

Corollary 7. $\forall i, j \in \mathcal{N}$, $LP_iP_j = \bar{P}_i\bar{P}_jL$.

Proof. The proof follows directly from point (2) of Lemma 9. □

In what follows, we consider two different sets of transformed error variables, i.e.,

$$\forall i \in \mathcal{N}, \qquad \bar{\boldsymbol{e}}_i(k) := L\boldsymbol{e}_i(k) \in \mathbb{R}^{\tilde{m}} \tag{5.73}$$

and

$$\forall i \in \mathcal{N}, \qquad \tilde{\boldsymbol{e}}_i(k) := \boldsymbol{e}_i(k) - L'\bar{\boldsymbol{e}}_i(k) \in \mathbb{R}^{m}. \tag{5.74}$$

Lemma 10. $\forall i \in \mathcal{N}$, $\bar{P}_i\bar{\boldsymbol{e}}_i(k) = \bar{\boldsymbol{e}}_i(k)$.

Proof. By (5.73), $\bar{P}_i\bar{\boldsymbol{e}}_i(k) = \bar{P}_iL\boldsymbol{e}_i(k)$. By property (2) of Lemma 9, $\bar{P}_iL\boldsymbol{e}_i(k) = LP_i\boldsymbol{e}_i(k)$, which, by Lemma 6, yields

$$\bar{P}_i\bar{\boldsymbol{e}}_i(k) = LP_i\boldsymbol{e}_i(k) = L\boldsymbol{e}_i(k) = \bar{\boldsymbol{e}}_i(k),$$

thus concluding the proof. □

Lemma 11. $\forall i, j \in \mathcal{N}$, $P_j\tilde{\boldsymbol{e}}_i(k) = \tilde{\boldsymbol{e}}_i(k)$.

Proof. Note that

$$L\tilde{\boldsymbol{e}}_i(k) = L\boldsymbol{e}_i(k) - LL'\bar{\boldsymbol{e}}_i(k) = L\boldsymbol{e}_i(k) - \bar{\boldsymbol{e}}_i(k) = 0,$$

since the columns of L are orthonormal. This implies $\tilde{\boldsymbol{e}}_i(k) \in \ker(L)$, which yields $\tilde{\boldsymbol{e}}_i(k) \in \bigcap_{j=1}^{\nu} \text{image}(P_j)$, then, $\forall i, j \in \mathcal{N}$, $P_j\tilde{\boldsymbol{e}}_i(k) = \tilde{\boldsymbol{e}}_i(k)$. This concludes the proof. □

Lemma 12. *If,* $\forall i \in \mathcal{N}$,

$$\lim_{k\to\infty} \bar{\boldsymbol{e}}_i(k) = \mathbf{0}_{\tilde{m}}$$

and

$$\lim_{k\to\infty} \tilde{\boldsymbol{e}}_i(k) = \boldsymbol{\epsilon}^*,$$

with $\boldsymbol{\epsilon}^* \in \mathbb{R}^m$, *then,* $\forall i \in \mathcal{N}$,

$$\lim_{k\to\infty} \boldsymbol{x}_i(k) = \boldsymbol{\epsilon}^* + x^*.$$

Proof. By merging (5.52) and (5.74), one obtains

$$\boldsymbol{x}_i(k) = \tilde{\boldsymbol{e}}_i(k) + L'\bar{\boldsymbol{e}}_i(k) + x^*.$$

The hypotheses of the Lemma yield

$$\lim_{k\to\infty} \boldsymbol{x}_i(k) = \lim_{k\to\infty} \tilde{\boldsymbol{e}}_i(k) + L'\bar{\boldsymbol{e}}_i(k) + x^* = \boldsymbol{\epsilon}^* + x^*,$$

which concludes the proof. □

By (5.53), (5.73), and Corollary 7, $\forall k \in \mathbb{N}_0$,

$$\bar{\boldsymbol{e}}_i(k+1) = \bar{P}_i \sum_{j=1}^{\nu} h_{ij}(k)\bar{\boldsymbol{e}}_j(k), \tag{5.75}$$

which in compact form becomes, $\forall k \in \mathbb{N}_0$,

$$\bar{\boldsymbol{e}}(k+1) = \bar{Q}(k)\bar{\boldsymbol{e}}(k), \tag{5.76}$$

where $\bar{Q}(k) := \text{blockdiag}\left(\bar{P}_1, \ldots, \bar{P}_\nu\right)(H(k) \otimes \mathbb{I}_m)$. Equations (5.76) and (5.56) are the same, apart from having $\bar{P}_i$ instead of P_i. By Lemma 9, $\bar{P}_i$ is also an orthogonal projection matrix, $i = 1 \ldots \nu$, and $\bigcap_{i=1}^{\nu} \text{image}(\bar{P}_i) = 0$. We can therefore repeat the argument from Section 5.2.4.1 to show that the mixed l_2/l_∞ norm of the product $\bar{Q}(k+\nu-1)\ldots\bar{Q}(k)$ is strictly smaller than 1, provided that $\mathfrak{G}$ is a repeatedly jointly $\mathcal{D}$-connected sequence of graphs and that the set of all $\bar{Q}(k)$ is finite (see Assumption 5 on pg. 76). Therefore, $\bar{\boldsymbol{e}}(k)$ converges exponentially to $\mathbf{0}_{\nu\tilde{m}}$, i.e., $\forall i \in \mathcal{N}$,

$$\lim_{k\to\infty} \bar{\boldsymbol{e}}_i(k) = \mathbf{0}_{\tilde{m}}. \tag{5.77}$$

Concerning $\tilde{\boldsymbol{e}}_i(k)$, by its definition,

$$\begin{aligned}\tilde{\boldsymbol{e}}_i(k+1) &= \boldsymbol{e}_i(k+1) - L'\bar{\boldsymbol{e}}_i(k+1)\\ &= P_i \sum_{j=1}^{\nu} h_{ij}(k)\boldsymbol{e}_j(k) - L'\bar{P}_i \sum_{j=1}^{\nu} h_{ij}(k)\bar{\boldsymbol{e}}_j(k),\end{aligned}$$

which, by property (3) in Lemma 9, becomes

$$\begin{aligned}&= P_i \sum_{j=1}^{\nu} h_{ij}(k)\boldsymbol{e}_j(k) - P_i L' \sum_{j=1}^{\nu} h_{ij}(k)\bar{\boldsymbol{e}}_j(k),\\ &= P_i \sum_{j=1}^{\nu} h_{ij}(k)\tilde{\boldsymbol{e}}_j(k).\end{aligned}$$

The latter, by Lemma 11, yields, $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$,

$$\tilde{\boldsymbol{e}}_i(k+1) = \sum_{j=1}^{\nu} h_{ij}(k)\tilde{\boldsymbol{e}}_j(k). \tag{5.78}$$

By Proposition 24 on pg. 79, and being $\mathfrak{G}$ also repeatedly jointly strongly connected, agents with dynamics (5.78) achieve consensus, namely, $\forall i \in \mathcal{N}$,

$$\lim_{k\to\infty} \tilde{\boldsymbol{e}}_i(k) = \boldsymbol{\epsilon}^*. \tag{5.79}$$

By Lemma 12, and both (5.77) and (5.79) having been verified under the hypotheses of the Theorem, all $\boldsymbol{x}_i(k)$ converge to the same solution. This shows that Theorem 8 is proven also in case of (5.47) having multiple solutions.

5.2.5 *Simulations*

In what follows, we simulate a set of agents trying to solve (5.47) by running algorithm (5.51) together with the designed communication protocol, under different conditions. For simplicity, we are considering fully connected underlying network topologies. Channel coefficients $\xi_{ij}(k)$ of each fully connected graph $\mathcal{G}(k)$ are drawn out of an uniform distribution[24], i.e., $\forall i,j \in \mathcal{N},\ \forall k \in \mathbb{N}_0,\ \xi_{ij}(k) \sim \mathcal{U}(0,\mu]$, where $\mu \in \mathbb{R}_{>0}$. In all simulations, $\nu = m$, thus, $\forall i \in \mathcal{N},\ n_i = 1$.

[24] They are independent and identically distributed.

5.2.5.1 *Unique Solution*

The matrices of equation (5.47) are the following:

$$A = \begin{pmatrix} \frac{22}{25} & \frac{61}{100} & \frac{31}{50} & \frac{1}{10} & \frac{8}{25} \\ \frac{27}{50} & \frac{3}{100} & \frac{19}{100} & \frac{49}{100} & \frac{23}{50} \\ \frac{1}{100} & \frac{8}{25} & \frac{81}{100} & \frac{8}{25} & \frac{19}{25} \\ \frac{87}{100} & \frac{8}{25} & \frac{22}{25} & \frac{18}{25} & \frac{53}{100} \\ \frac{13}{100} & \frac{41}{100} & \frac{59}{100} & \frac{57}{100} & \frac{3}{25} \end{pmatrix}, \qquad B = \begin{pmatrix} \frac{93}{100} \\ \frac{17}{50} \\ \frac{18}{25} \\ \frac{19}{50} \\ \frac{47}{100} \end{pmatrix}.$$

This is the case of (5.47) having a unique solution, namely $n = m$ and A full-row rank. Fig. 5.23 represents the evolution of $x_i(k)$, $i = 1 \dots \nu$, through iterations. The evolution of $\|\boldsymbol{e}(k)\|_{2,\infty}$ through iterations can be seen in Fig. 5.24. In the same figure, we have plotted also $\log(\|\boldsymbol{e}(k)\|_{2,\infty})$, thus showing the exponential convergence to a solution.

5.2.5.2 *Ideal Channel*

The impact of the unknown fading channel can be addressed by looking at Fig. 5.25, where uncertainty and fading are removed, namely, $\forall i,j \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $\xi_{ij}(k) = 1$. Note that this can only be done using *standard* orthogonal channel access methods, i.e., at every iteration step, all agents would have to transmit the current estimations independently. Hence, if this is done, e.g., by TDMA (Time-division multiple access), we would need νm orthogonal transmissions per iteration. The comparison of Fig. 5.23 and Fig. 5.25 illustrates that the

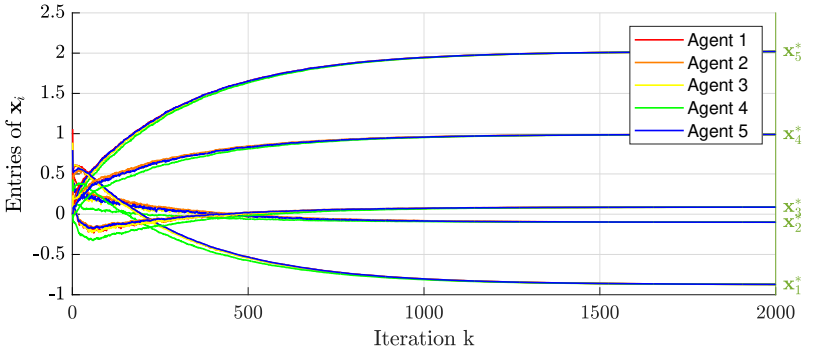


Figure 5.23: Convergence to the solution for (5.47) having an unique solution.

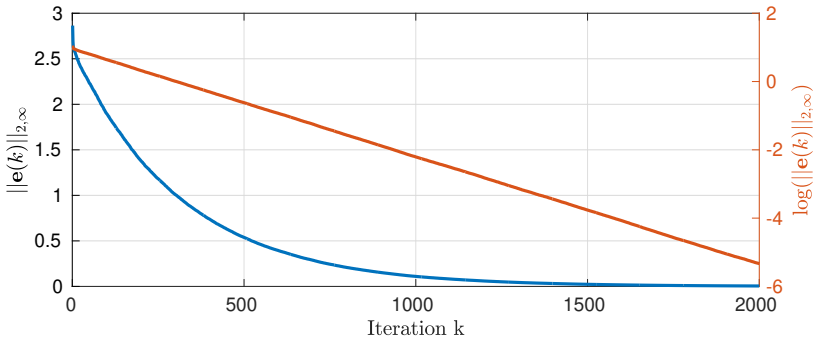


Figure 5.24: $\|\boldsymbol{e}(k)\|_{2,\infty}$ and $\log(\|\boldsymbol{e}(k)\|_{2,\infty})$ through iterations.

presence of an unknown and fading channel does not have a noticeable impact.

Comparing Fig. 5.23 with Fig. 5.25 helps quantifying the savings of wireless resources in case interference is exploited. In fact, although Fig. 5.23 and Fig. 5.25 exhibit the same convergence rate, the traditional communication approach (in which agents access the channel separately) requires at least $m\nu$ independent channel accesses per every iteration. In fact, each agent $(1 \dots \nu)$ separately communicates to neighbors each entry $(1 \dots m)$ of its information state. Our designed communication system requires only $m + 1$, but guarantees the same convergence rate in terms of iterations as the traditional approach. Exploiting interference, therefore, significantly reduces the consumption of wireless resources.

5.2.5.3 *Multiple Solution*

The matrices of equation (5.47) are the following:

$$A = \begin{pmatrix} \frac{22}{25} & \frac{61}{100} & \frac{31}{50} & \frac{1}{10} & \frac{8}{25} \\ \frac{27}{50} & \frac{3}{100} & \frac{19}{100} & \frac{49}{100} & \frac{23}{50} \\ \frac{1}{100} & \frac{8}{25} & \frac{81}{100} & \frac{8}{25} & \frac{19}{25} \\ \frac{87}{100} & \frac{8}{25} & \frac{22}{25} & \frac{18}{25} & \frac{53}{100} \end{pmatrix}, \qquad B = \begin{pmatrix} \frac{93}{100} \\ \frac{17}{50} \\ \frac{18}{25} \\ \frac{19}{50} \end{pmatrix}$$

The numerical simulation is depicted by Fig. 5.26.

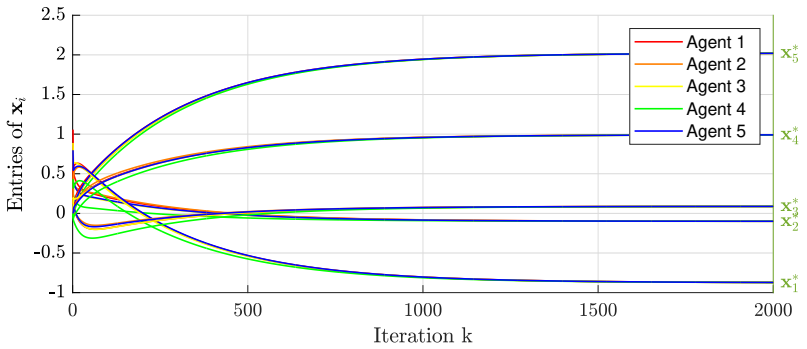


Figure 5.25: $x_i(k)$ through iterations, with an ideal channel.

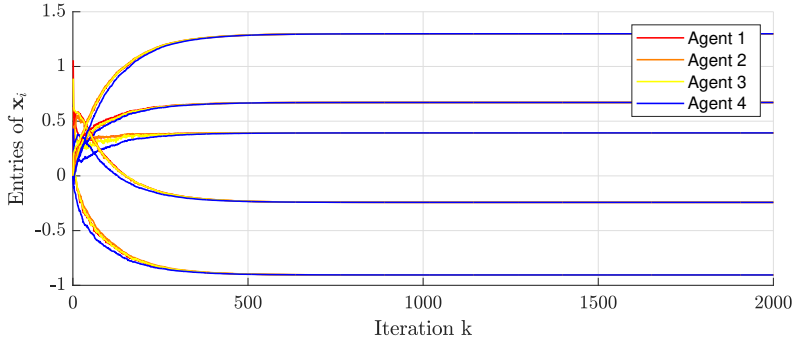


Figure 5.26: $x_i(k)$ through iterations for the case of multiple solutions.

6
Max Consensus Over the Wireless Channel

Partial results of the presented section have been published in Molinari et al. [64] and Molinari et al. [68]. The candidate is the first author of all these contributions.

As introduced in Section 2.3.2, *max-consensus* protocols allow agents to reach an agreement on the largest information state. Standard max-consensus protocols guarantee that the agreement on the maximum value is achieved in finite time, see, e.g., [33, 75]. Since this technology finds application in contexts where converging time is a critical resource (see, e.g., Section 3), any method that can diminish such a quantity is of extreme interest for research. As for the average consensus (Section 5), and motivated by Section 1, the goal of this section is designing a consensus algorithm that exploits the superposition property of the wireless channel and guarantees convergence to the max-consensus.

Note that, throughout this section, the underlying network topology is time-invariant strongly connected, i.e., $\mathcal{G} = (\mathcal{N}, \mathcal{A})$. This section designs a max-consensus protocol based on the interference model of Section 4.2.2.

6.1 Asymptotic Converging Algorithm

Let $\mathcal{N}$ denote the set of agents, labeled 1 through $n \in \mathbb{N}$, exchanging information over a network modeled by a time-invariant strongly connected directed graph, i.e., $\mathcal{G} = (\mathcal{N}, \mathcal{A})$. Let, $\forall i \in \mathcal{N},\ \forall k \in \mathbb{N}_0,\ x_i(k)$ denote the information state of agent i at iteration k. Vector $\boldsymbol{x}(k) \in \mathbb{R}^n$ stacks all information states at iteration $k \in \mathbb{N}_0$. The wireless channel is modeled by a fading noiseless WMAC, see Chapter 4. Let us initially assume that the communication system of Section 5.1.1 is employed also here; thus, the signal obtained by each agent $i \in \mathcal{N}$ at every iteration $k \in \mathbb{N}_0$ is

$$u_i(k) := \sum_{j \in \mathcal{N}_i(k)} h_{ij}(k) x_i(k), \tag{6.1}$$

where $h_{ij}(k)$ defined as in (5.8).

Definition 23. *Any agent, say $i \in \mathcal{N}$, whose information state at iteration $k \in \mathbb{N}_0$ is $x_i(k) = \bar{x} = \max_{i \in \mathcal{N}} x_i(0)$ is referred to as **maximal agent** at iteration $k \in \mathbb{N}_0$.*

The proposed max-consensus protocol is primarily based on two underlying ideas, summarized in Observation 3 and Proposition 27.

Observation 3. *If the goal is to achieve max-consensus, any non-maximal agent at iteration $k \in \mathbb{N}_0$ does not need to communicate its own information state at iteration $k \in \mathbb{N}_0$.*

However, agents in general do not know whether they are maximal at a given iteration k. Agents only have a local estimation of this. Let us assume that the result of this local evaluation for agent $i \in \mathcal{N}$ at iteration $k \in \mathbb{N}_0$ is stored in a binary variable $\tilde{y}_i(k) \in \{0, 1\}$. In the case $\tilde{y}_i(k) = 1$, agent $i \in \mathcal{N}$ is said to be a **maximal-candidate** at iteration k. If (and only if) agent i is a maximal-candidate, it will be allowed to broadcast at the next iteration. This will be expressed by an **authorization variable** $y_i : \mathbb{N}_0 \to \{0, 1\}$, where $y_i(k) = \tilde{y}_i(k-1)$ and $y_i(0) = 1$.

Now, let,

$$\forall i \in \mathcal{N},\ \forall k \in \mathbb{N}_0,$$
$$N_i^m(k) := \{j \in N_i \mid y_j(k) = 1\} \subseteq N_i \quad (6.2)$$

be the set of neighbors authorized to broadcast at iteration k. If only authorized agents broadcast and the communication system of Section 5.1.1 is employed, each agent $i \in \mathcal{N}$ will receive, at $k \in \mathbb{N}_0$,

$$u_i(k) = \sum_{j \in N_i^m(k)} h_{ij}(k) x_j(k). \quad (6.3)$$

The local evaluation that establishes which agents are authorized to broadcast is based on the following proposition.

Proposition 27. *Given a set of agents $\mathcal{N}$, a non-empty subset $\mathcal{M} \subseteq \mathcal{N}$, and a set of real-valued parameters $\mathcal{H} = \{h_j \in (0, 1] \mid j \in \mathcal{M}\}$ with $\sum_{j \in \mathcal{M}} h_j = 1$, the following holds $\forall k \in \mathbb{N}_0$, $\forall i \in \mathcal{N}$,*

$$x_i(k) < \sum_{j \in \mathcal{M}} h_j x_j(k) \implies x_i(k) < \max_{j \in \mathcal{N}}(x_j(k)). \quad (6.4)$$

Proof. By definition of a convex hull, $\sum_{j \in \mathcal{M}} h_j x_j(k) \in \mathcal{C}(\{x_j(k) \mid j \in \mathcal{M}\})$. Moreover, $\forall p \in \mathcal{C}(\{x_j(k) \mid j \in \mathcal{M}\}) : p \leq \max_{j \in \mathcal{M}}(x_j(k))$. Hence,

$$\sum_{j \in \mathcal{M}} h_j x_j(k) \leq \max_{j \in \mathcal{M}}(x_j(k)).$$

Since $\mathcal{M} \subseteq \mathcal{N}$,

$$x_i(k) < \max_{j \in \mathcal{M}}(x_j(k)) \leq \max_{j \in \mathcal{N}}(x_j(k)).$$

This implies (6.4). □

By (6.4), for $\mathcal{M} = N_i^m(k)$ and $\mathcal{H} = \{h_{ij}(k) \mid j \in N_i^m(k)\}$, the implication

$$x_i(k) < u_i(k) \implies x_i(k) < \max_{j \in \mathcal{N}}(x_j(k)) \quad (6.5)$$

immediately follows. Therefore, y_i can be updated as

$$\forall i \in \mathcal{N},\ \forall k \in \mathbb{N}_0,$$
$$y_i(k+1) = \tilde{y}(k) = I_{\mathbb{R}_{\geq 0}}(x_i(k) - u_i(k)). \quad (6.6)$$

In the light of these observations, and given that the signal $u_i(k)$ is computed by harnessing the interference of the channel, each agent $i \in \mathcal{N}$ can apply the following max-consensus protocol:

$$\forall k \in \mathbb{N}_0,\ \begin{cases} x_i(k+1) = \max(x_i(k), u_i(k)) \\ y_i(k+1) = I_{\mathbb{R}_{\geq 0}}(x_i(k) - u_i(k)) \end{cases}, \quad (6.7)$$

where $y_i(0) = 1$ and $x_i(0) = x_{i_0} \in \mathsf{S}$, and $u_i(k)$ is obtained from (6.3). Note that $u_i(k)$ is determined by $x_j(k)$ and $y_j(k)$, $j \in N_i$. (6.3)-(6.7) can then be rewritten in vector-form as

$$\mathbf{w}(k+1) = g(\mathbf{w}(k)), \quad (6.8)$$

where

$$\mathbf{w}(k) = \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{y}(k) \end{bmatrix}, \quad (6.9)$$

and, $\forall i \in \mathcal{N},\ [\mathbf{x}(k)]_i = x_i(k),\ [\mathbf{y}(k)]_i = y_i(k)$, and $g : \mathbb{R}^n \times \{0,1\}^n \to \mathbb{R}^n \times \{0,1\}^n$ is the nonlinear function reflecting (6.7) and (6.3).

6.1.1 *Asymptotic Convergence*

A multi-agent system with a strongly connected network topology $(\mathcal{N}, \mathcal{A})$ is given. The system uses the consensus protocol (6.8), i.e., each agent iterates (6.3),(6.7) synchronously. In the following, we prove asymptotic convergence by using Lyapunov theory (cf. [6, p. 87] and [46, p. 22]). Initially, we show that all information states are non-decreasing bounded sequences.

Proposition 28. *Given a multi-agent system with network topology* $(\mathcal{N}, \mathcal{A})$ *and consensus protocol* (6.8), $\forall \mathbf{x}(0) \in \mathsf{S}^n$, $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$,

$$x_i(k) \leq x_i(k+1) \leq \max_{j \in \mathcal{N}}(x_j(0)). \quad (6.10)$$

Proof. The first inequality immediately follows from (6.7). The second inequality follows from the fact that, according to (6.3),

$$u_i(k) \in \mathcal{C}\left(\{x_j(k) \mid N_i^m(k)\}\right)$$

if $N_i^m(k) \neq \emptyset$, zero else. Hence

$$u_i(k) \leq \max_{j \in N_i^m(k)}\left(x_j(k)\right) \leq \max_{j \in \mathcal{N}}\left(x_j(k)\right).$$

Therefore, $\forall i \in \mathcal{N}$,

$$x_i(k+1) \leq \max_{j \in N_i^m(k)}\left(x_j(k)\right) \leq \max_{j \in \mathcal{N}}\left(x_j(k)\right).$$

Moreover,

$$\max_{j \in \mathcal{N}}\left(x_j(k+1)\right) \leq \max_{j \in \mathcal{N}}\left(x_j(k)\right),$$

thus yielding the second inequality. □

The following propositions establish a unique equilibrium point.

Proposition 29.

$$\mathbf{w}^* = \left[\mathbf{x}^{*\prime}, \mathbf{1}_n'\right]', \tag{6.11}$$

with $\mathbf{x}^* = x^*\mathbf{1}_n$ *and* $x^* = \max_{j\in\mathcal{N}}(x_j(0))$ *is an equilibrium point for the multi-agent system with network topology* $(\mathcal{N}, \mathcal{A})$ *and consensus protocol* (6.8).

Proof. Assume $\mathbf{w}(k) = \mathbf{w}^*$. This implies, $\forall i \in \mathcal{N}$, $x_i(k) = x^*$ and $y_i(k) = 1$. Therefore $N_i^m(k) = N_i$, hence

$$\begin{aligned} u_i(k) &= \sum_{j\in N_i} h_{ij}(k)x_j(k) \\ &= \left(\sum_{j\in N_i} h_{ij}(k)\right) x^* = x^*, \end{aligned} \tag{6.12}$$

as $N_i \neq \emptyset$. Then, according to (6.7),

$$x_i(k+1) = \max(x^*, u_i(k)) = x^* \tag{6.13}$$

and

$$y_i(k+1) = I_{\mathbb{R}_{\geq 0}}(x^* - u_i(k)) = 1, \tag{6.14}$$

and, therefore, $\mathbf{w}(k+1) = \mathbf{w}^*$. □

Proposition 30. *Consider a multi-agent system with a strongly connected network topology* $(\mathcal{N}, \mathcal{A})$ *using the consensus protocol* (6.7).

$$\mathbf{w}^* = [x^*\mathbf{1}_n', \mathbf{1}_n']'$$

is the unique equilibrium point.

Proof. The proof is by contradiction, i.e., we assume that there exists an equilibrium point

$$\hat{\mathbf{w}} = [\hat{\mathbf{x}}', \hat{\mathbf{y}}']' \neq \mathbf{w}^*.$$

Case 1

$\hat{\mathbf{y}} \neq \mathbf{1}_n'$, i.e.,

$$\exists i \in \mathcal{N}, \text{ s.t. } \hat{\mathbf{y}}_i = 0.$$

Hence, because of (6.7),

$$x_i(k) < u_i(k)$$

and therefore $x_i(k+1) > x_i(k)$. Hence we have established that for any equilibrium point $\hat{\mathbf{w}}$ the Boolean part needs to be

$$\hat{\mathbf{y}} = \mathbf{1}_n'.$$

Case 2

$\hat{\mathbf{y}} = \mathbf{1}'_n$. but $\hat{\mathbf{x}} \neq x^* \mathbf{1}'_n$. Because of the first premise, $\forall i \in \mathcal{N}$,

$$N_i^m(k) = N_i.$$

As $(\mathcal{N}, \mathcal{A})$ is strongly connected, $N_i \neq \emptyset$, $\forall i \in \mathcal{N}$. Furthermore, also because of $(\mathcal{N}, \mathcal{A})$ being strongly connected, there exists a minimal element $l \in \mathcal{N}$ that has at least one non-minimal neighbor, i.e.,

$$\hat{x}_l = \min_{j \in \mathcal{N}} \hat{x}_j$$

and

$$\hat{x}_l < \max_{j \in N_l} \hat{x}_j := \hat{x}_p.$$

As the channel coefficients are positive, (6.3) implies

$$u_l(k) > x_l(k).$$

This and (6.7) imply

$$x_l(k+1) > x_l(k).$$

Hence, we have established that for any equilibrium point $\hat{\mathbf{w}}$, the real part has to be $x^* \mathbf{1}_n$. □

Lemma 13. *Consider a multi-agent system with a strongly connected network topology $(\mathcal{N}, \mathcal{A})$. If protocol (6.8) is employed, then, $\forall \mathbf{x}(0) \in \mathrm{S}^n$, $\forall k \in \mathbb{N}_0$,*

$$\sum_{i \in \mathcal{N}} (x_i(k+2) - x_i(k)) = 0 \implies \mathbf{x}(k) = \mathbf{x}^*. \tag{6.15}$$

Proof. From Proposition 28, $\{x_i(k)\}_{k \in \mathbb{N}_0}$ is a non-decreasing bounded sequence, composed of nonnegative entries. As a consequence,

$$\sum_{i \in \mathcal{N}} (x_i(k+2) - x_i(k)) = 0$$

if and only if

$$\mathbf{x}(k) = \mathbf{x}(k+1) = \mathbf{x}(k+2). \tag{6.16}$$

The latter, by (6.7), implies that, $\forall i \in \mathcal{N}$, $x_i(k) \geq u_i(k)$ and $x_i(k+1) \geq u_i(k+1)$. By (6.6), this implies that

$$\mathbf{y}(k+1) = \mathbf{y}(k+2) = 1. \tag{6.17}$$

From (6.16) and (6.17), it clearly follows that

$$\mathbf{w}(k+1) = \begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{y}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{x}(k+2) \\ \mathbf{y}(k+2) \end{bmatrix} = \mathbf{w}(k+2) \tag{6.18}$$

is an equilibrium for the system. According to Proposition 30, it is unique, i.e.,

$$\mathbf{w}(k+1) = \mathbf{w}(k+2) = \mathbf{w}^*, \tag{6.19}$$

$$\mathbf{x}(k+1) = \mathbf{x}(k+2) = \mathbf{x}^*. \tag{6.20}$$

By (6.16), $\mathbf{x}(k) = \mathbf{x}^*$; this concludes the proof. □

Corollary 8. *Consider a multi-agent system with a strongly connected network topology* $(\mathcal{N}, \mathcal{A})$*. If the protocol* (6.8) *is employed, then,* $\forall \mathbf{x}(0) \in \mathrm{S}^n$, $\forall k \in \mathbb{N}_0$,

$$\mathbf{x}(k) \neq \mathbf{x}^* \implies \sum_{i \in \mathcal{N}} (x_i(k+2) - x_i(k)) > 0. \tag{6.21}$$

Proof. (6.15) is equivalent to

$$\mathbf{x}(k) \neq \mathbf{x}^* \implies \sum_{i \in \mathcal{N}} (x_i(k+2) - x_i(k)) \neq 0.$$

Non-decreasingness of the sequence $\{x_i(k)\}_{k \in \mathbb{N}_0}$ (Proposition 28) then establishes (6.21). □

By [108, p. 264] and [74, p. 43], a Lyapunov-based analysis can be applied to the discrete-time system like (6.8), as existence and uniqueness of an equilibrium point have been established in Proposition 29 and Proposition 30.

Theorem 10. *Consider a multi-agent system with a strongly connected network topology* $(\mathcal{N}, \mathcal{A})$*. Agents employ the consensus protocol* (6.8)*. For every possible initial state* $\mathbf{x}(0) \in \mathrm{S}^n$*, the system achieves max-consensus asymptotically.*

Proof. The component y_i of (6.7) can be explicitly rewritten, $\forall k \in \mathbb{N}_0$, as

$$y_i(k+1) = \begin{cases} 1 & \text{if } x_i(k) \geq u_i(k) \\ 0 & \text{if } x_i(k) < u_i(k) \end{cases}. \tag{6.22}$$

By (6.7) and since $\{x_i(k)\}_{k \in \mathbb{N}_0}$ is a non-decreasing sequence, (6.22) can be reformulated, $\forall k \in \mathbb{N}_0$, as

$$y_i(k+1) = \begin{cases} 1 & \text{if } x_i(k) = x_i(k+1) \\ 0 & \text{if } x_i(k) < x_i(k+1) \end{cases}. \tag{6.23}$$

This, again because of non-decreasingness of $\{x_i(k)\}_{k \in \mathbb{N}_0}$, is equivalent to

$$\forall k \in \mathbb{N}_0,\ y_i(k+1) = I_{\mathbb{R}_{\geq 0}}(x_i(k) - x_i(k+1)). \tag{6.24}$$

By introducing the new state vector

$$\mathbf{v}(k) := \begin{bmatrix} \mathbf{v}_1(k) \\ \mathbf{v}_2(k) \end{bmatrix} := \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{x}(k-1) \end{bmatrix}. \tag{6.25}$$

we can rewrite (6.8) as

$$\mathbf{v}(k+1) = \tilde{g}(\mathbf{v}(k)), \tag{6.26}$$

where the function $\tilde{g} : S^{2n} \mapsto S^{2n}$ can be explicitly expressed by

$$\mathbf{v}_1(k+1) = \max(\mathbf{v}_1(k), H(k)\mathbf{v}_1(k)) \tag{6.27}$$

$$\mathbf{v}_2(k+1) = \mathbf{v}_1(k), \tag{6.28}$$

where $\max(\cdot)$ is understood element-wise and

$$[H(k)]_{ij} := \begin{cases} h_{ij}(k) & \text{if } j \in N_i \text{ and } [\mathbf{v}_1(k)]_j = [\mathbf{v}_2(k)]_j \\ 0 & \text{else} \end{cases}.$$

Consequently, the unique equilibrium $\mathbf{w}^*$ of (6.8) corresponds to the following unique equilibrium $\mathbf{v}^*$ of (6.26):

$$\mathbf{v}^* = x^* \mathbf{1}_{2n}. \tag{6.29}$$

A candidate Lyapunov function $V : \mathrm{S}^{2n} \to \mathbb{R}_{\geq 0}$ is chosen as follows:

$$\forall k \in \mathbb{N}_0,\ V(\mathbf{v}(k)) = \mathbf{1}_{2n}(\mathbf{v}^* - \mathbf{v}(k)) \tag{6.30}$$

$$= 2nx^* - \sum_{i \in \mathcal{N}} (x_i(k) + x_i(k-1)). \tag{6.31}$$

The following properties hold:

a) $V(\mathbf{v})$ is continuous in $\mathbb{R}^{2n}_{\geq 0}$;

b) $V(\mathbf{v}^*) = 0$;

c) $V(\mathbf{v})$ is positive on any trajectory $\mathbf{v}$ of the system, unless $\mathbf{v} = \mathbf{v}^*$.
This follow directly from Proposition 28.

d) By Corollary 8, $\forall \mathbf{v}(k) \neq \mathbf{v}^*$ (i.e., $\forall \mathbf{x}(k-1) \neq \mathbf{x}^*$),

$$\begin{aligned} \Delta V(\mathbf{v}(k)) &= V(\mathbf{v}(k+1)) - V(\mathbf{v}(k)) \\ &= -\sum_{i \in \mathcal{N}} (x_i(k+1) - x_i(k-1)) < 0. \end{aligned}$$

Hence, the function $V(\mathbf{v})$ has all the properties required for a Lyapunov function. By [46, p. 22] and [6, p. 88], the system therefore asymptotically converges to max-consensus. □

In the following corollary, an immediate result coming as a consequence from Theorem 10 is reviewed. It will be used in the next section.

Corollary 9. *Consider a multi-agent system with a strongly connected network topology $(\mathcal{N}, \mathcal{A})$ iterating consensus protocol (6.8).*

$$\forall i \in \mathcal{N},\ x_i(k_0) < x^* \implies \exists k_i > k_0 :\ y_i(k_i) = 0. \tag{6.32}$$

Proof. Let $x^* - x_i(k_0) = \epsilon_0$. Choose $\epsilon < \epsilon_0$. Asymptotic stability implies that $\exists k_\epsilon > k_0$ such that $x^* - x_i(k) < \epsilon$, $\forall k > k_\epsilon$. Hence, $x_i(k) > x_i(k_0)$, $\forall k > k_\epsilon$. This implies that $\exists k_i$, $k_0 < k_i \leq k_\epsilon$ s.t.

$$x_i(k_i) - x_i(k_i - 1) > 0,$$

therefore $y_i(k_i) = 0$. □

The interpretation of this corollary is that an agent which is non-maximal at iteration k_0, will eventually lose its status as a maximal candidate and therefore its authorization to broadcast.

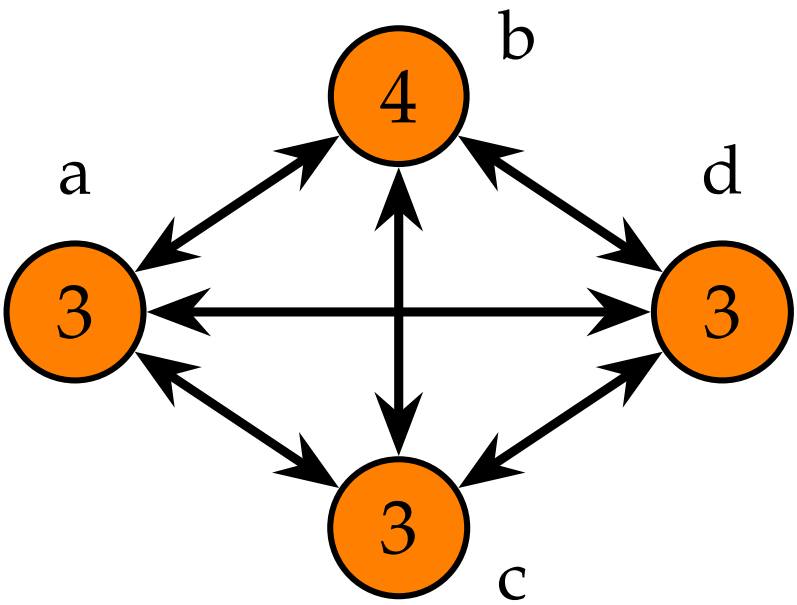


Figure 6.1: Network topology for Examples 1, 2, 4. The node set is $\{a, b, c, d\}$ and the vector of initial information states is $\mathbf{x}(0) = [3, 4, 3, 3]'$ for Example 1 and 4, and $\mathbf{x}(0) = [3.1, 4, 3, 3]'$ for Example 2.

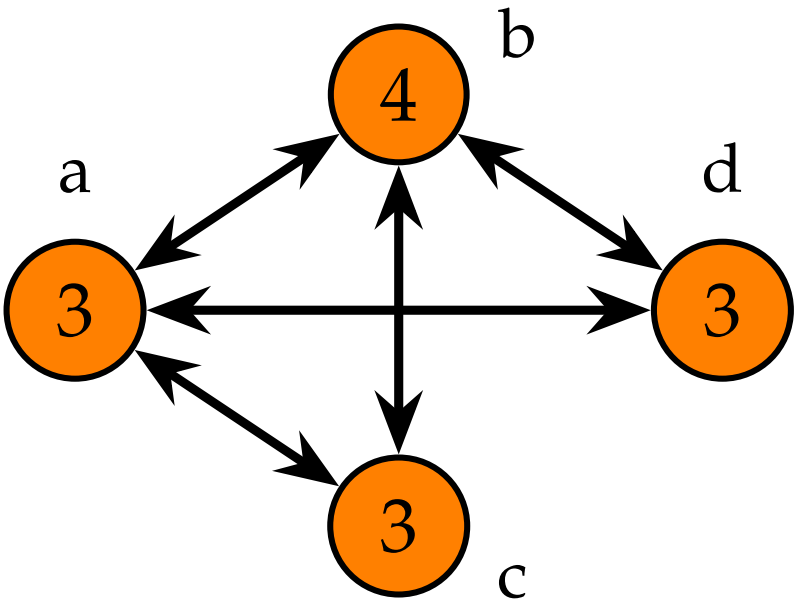


Figure 6.2: Network topology for Example 3. The node set is $\{a, b, c, d\}$ and the vector of initial information states is $\mathbf{x}(0) = [3, 4, 3, 3]'$.

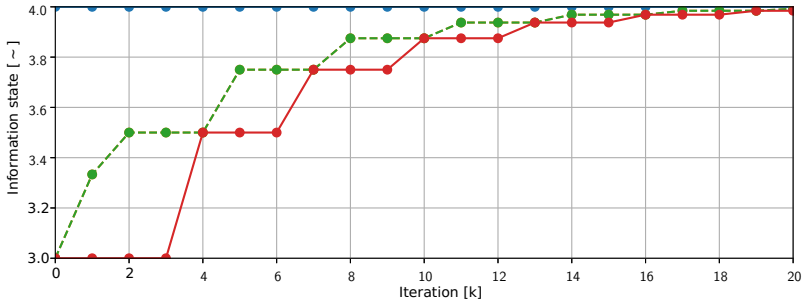


Figure 6.3: Evolution of agents' information states for Example 1. The dashed lines represent $x_a(k)$ and $x_d(k)$ (which coincide). The red solid line represents the information state of agent c. Asymptotic convergence to the max value, i.e., $x^* = x_b(0)$, can be observed.

In conclusion, we have shown that a multi-agent system with a strongly connected network topology achieves max-consensus asymptotically by employing protocol (6.8). This protocol harnesses the interference property of the channel for computing the signal u_i. In the following, we show simulation studies that provide further information on convergence and computational benefits of the suggested approach.

6.1.2 *Simulations*

In the following, some simulation results are presented. The aim is to compare our approach to standard approaches and quantify benefits (saved wireless resources). However, as it will be clear in the simulations, there are cases in which finite-time convergence cannot be achieved with protocol (6.8), but only asymptotic convergence. In such cases, we cannot quantify benefits. This will motivate the search for an extended protocol in the next section.

Example 1: first, we review a numerical experiment in [64], where channel coefficients are equal and constant. Consider the underlying network in Figure 6.1-6.2. By (6.3), $u_i(k)$ is the linear average of information states of agents in $N_i^m(k)$. Example 1 illustrates asymptotic convergence of a system composed of 4 nodes, with a strongly connected network topology, endowed with protocol (6.8), see Figure 6.3.

Examples 2-3: on the other hand, Examples 2, respectively Example 3, show that, by slightly varying $\mathbf{x}(0)$, respectively the network topology, the system achieves finite-time max-consensus (see Figures 6.4 and 6.5). This behavior has been confirmed by running extensive numer-

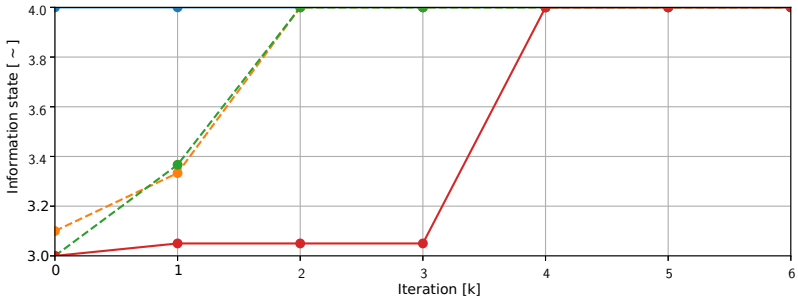


Figure 6.4: Evolution of agents' information states for Example 2. The dashed lines represent the information states of agents a and d, the solid line that of agent c. Max-consensus $x_b(0)$ is achieved after 4 iterations.

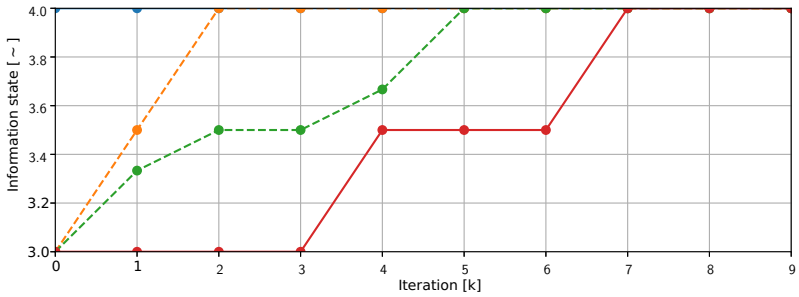


Figure 6.5: Evolution of agents' information states for Example 3. The dashed lines represent the information states of agents a and d, the solid line that of agent c. Max-consensus $x_b(0)$ is achieved after 7 iterations.

ical simulations: in most cases, finite time convergence, rather than asymptotic convergence, is achieved.

Examples 4: (see Figure 6.6), channel coefficients are randomly drawn out of a Rayleigh distribution with variance 1 independently for every iteration. Numerical experiments indicate that the system is *very likely* to achieve finite-time max-consensus. However, it will be possible to choose a collection of constant channel coefficients, i.e. $\forall k \in \mathbb{N}_0,\ h_{ji}(k) = h$, so that consensus is achieved asymptotically, rather than finite-time.

These examples illustrate that the use of protocol (6.7) for a strongly connected network does not guarantee finite-time consensus. Asymptotic convergence is guaranteed by Theorem 10; the achievement of finite-time consensus, on the other hand, depends on the network topology, the initial information states, and the channel coefficients if protocol (6.7) is used. This is the motivation for establishing an extended max-consensus protocol in the next section.

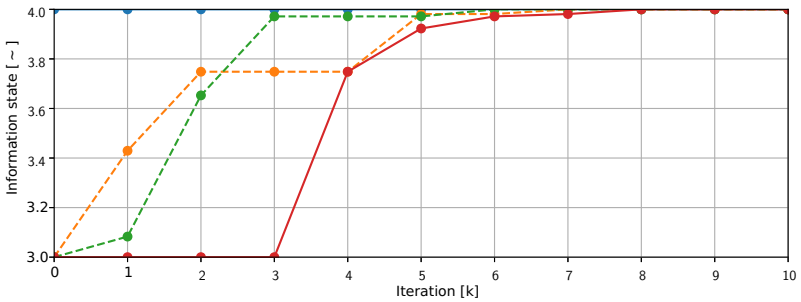


Figure 6.6: Evolution of agents' information states in the case of a fading wireless channel. This has an important impact on the convergence: with regards to Figure 6.3 (nonfading channel), convergence is here achieved in a finite number of steps, rather than asymptotically.

6.2 Finite-time Max-Consensus Protocol

In this section, an extended max-consensus protocol that exploits the channel superposition property is presented. However, in contrast to the protocol (6.8), it guarantees finite-time convergence for strongly connected network topologies.

6.2.1 Key idea

Let, $\forall k \in \mathbb{N}_0$, $\mathcal{M}_k$ be the set of maximal-agents at iteration k, i.e. $\mathcal{M}_k = \{i \in \mathcal{N} \mid x_i(k) = x^*\}$. The following proposition states that, at any iteration, there exists a non-maximal agent if and only if there is a non-maximal agent in whose neighborhood there is a maximal agent.

Proposition 31. *Given a multi-agent system with a strongly connected network topology $(\mathcal{N}, \mathcal{A})$, then*

$$\exists j \in \mathcal{N} \setminus \mathcal{M}_k \iff \exists i \in \mathcal{N} \setminus \mathcal{M}_k \text{ such that } N_i \cap \mathcal{M}_k \neq \emptyset$$

Proof.

$\Longleftarrow$ Trivial.

$\Longrightarrow$ Partition $\mathcal{N}$ as $\mathcal{M}_k$ and $\mathcal{N} \setminus \mathcal{M}_k$. Choose $j \in \mathcal{N} \setminus \mathcal{M}_k$ and $\bar{j} \in \mathcal{M}_k$. Because of strong connectedness, there is a path from $\bar{j}$ to j. Clearly, there is at least one arc in this path, say $(\bar{i}, i)$, such that $\bar{i} \in \mathcal{M}_k$ and $i \in \mathcal{N} \setminus \mathcal{M}_k$. As $\bar{i}$ is a neighbour of i, then $\bar{i} \in N_i \cap \mathcal{M}_k$.

□

The following result is derived directly from Corollary 9.

Proposition 32. *Given a multi-agent system with a strongly connected network topology $(\mathcal{N}, \mathcal{A})$ endowed with the consensus protocol (6.8), given an arbitrary $k_0 \in \mathbb{N}_0$ and $\forall \mathbf{x}(k_0) \in \mathrm{S}^n$, the following holds:*

$$\exists \tilde{k} > k_0 : \ \forall i \in \mathcal{N} \setminus \mathcal{M}_{k_0}, \ \prod_{t=k_0}^{\tilde{k}} y_i(t) = 0. \tag{6.33}$$

Proof. Corollary 9 states that for each $i \in \mathcal{N} \setminus \mathcal{M}_{k_0}$, there exists $k_i > k_0$ such that $y_i(k_i) = 0$. Take

$$\tilde{k} := \max_{i \in \mathcal{N} \setminus \mathcal{M}_{k_0}} k_i.$$

Then, $\forall i \in \mathcal{N} \setminus \mathcal{M}_{k_0}$,

$$\prod_{t=k_0}^{\tilde{k}} y_i(t) = 0.$$

The proof is concluded. □

By Proposition 32, each agent of the system, say agent i, that at k_0 is not maximal (i.e. $x_i(k_0) < x^*$), within $(\tilde{k} - k_0)$ steps will receive an input $u_i(k)$, $k \in [k_0, \tilde{k} - 1]$, such that $u_i(k) > x_i(k)$.

Now suppose that we change the consensus protocol (6.7) by setting the authorization variable of each agent $i \in \mathcal{N}$ at iteration $\tilde{k}+1$ to

$$y_i(\tilde{k}+1) = \prod_{t=k_0}^{\tilde{k}} y_i(t). \tag{6.34}$$

From Proposition 32, this quantity is zero for all agents that were non-maximal at k_0, implying that

$$\forall i \in \mathcal{N},\ N_i \cap \mathcal{M}_{k_0} \neq \emptyset \implies x_i(\tilde{k}+2) = x^*. \tag{6.35}$$

In other words, all agents in the neighborhood of a maximal agent will become maximal at iteration $\tilde{k}+2$.

However, since agents do not have the global knowledge of the system, the value of $\tilde{k}$ is not known a priori. Hence, there will be the need for each agent $i \in \mathcal{N}$ to retain a state variable, say $T_i : \mathbb{N}_0 \to \mathbb{N}$, that attempts to (over-)estimate $\tilde{k}$. By letting $T_i(k)$ grow according to a nondecreasing diverging sequence, it will be eventually large enough to over-approximate $\tilde{k}$.

6.2.2 *Protocol Design*

The idea just presented inspires the following switching consensus protocol $\forall i \in \mathcal{N}, \forall k \in \mathbb{N}_0$,

if $k = 2T_i(k)$:

$$\begin{cases} x_i(k+1) = \max(x_i(k), u_i(k)) \\ y_i(k+1) = \prod_{t=T_i(k)}^{k} y_i(t) \\ T_i(k+1) = k \end{cases}, \tag{6.36a}$$

else:

$$\begin{cases} x_i(k+1) = \max(x_i(k), u_i(k)) \\ y_i(k+1) = I_{\mathbb{R}_{\geq 0}}(x_i(k) - u_i(k)) \\ T_i(k+1) = T_i(k) \end{cases}, \tag{6.36b}$$

where $\forall i \in \mathcal{N}$, $y_i(0) = 1$, $x_i(0) = x_{i_0}$, $T_i(0) = T(0) = 2$, and $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $u_i(k)$ is computed as in (6.3), by exploiting the superposition property of the channel. Protocol (6.36b) is identical to protocol (6.7), except for the trivial presence of $T_i(k)$, which is, however, kept constant and does not affect the system behavior. Only for iteration steps $k = 2^n,\ n \in \mathbb{N}$, the proposed consensus protocol switches to (6.36a).

Proposition 33. $\forall i \in \mathcal{N}, \forall k \in \mathbb{N}_0$,

$$T_i(k) = 2^{p(k)}, \tag{6.37}$$

where

$$p(k) := \begin{cases} \lceil \log_2(k) - 1 \rceil & \text{if } k \geq 2 \\ 1 & \text{else} \end{cases}. \tag{6.38}$$

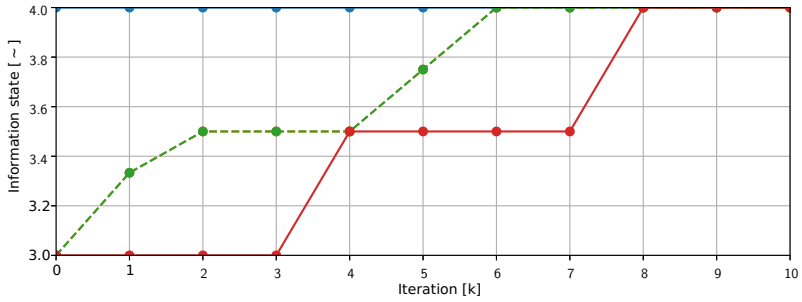


Figure 6.7: Evolution of information states through iterations in the absence of a fading channel. The solid line indicates the information state of agent c, whilst the dashed lines (overlapped) are the ones of agents a and d. Max-consensus is achieved in 8 iterations.

Proof. Follows directly from (6.36). □

Proposition 33 implies that $T_i(k)$, $k \in \mathbb{N}_0$, is a non-decreasing diverging sequence.

Remark 11. *The state variable $T_i(k)$ is the same for all $i \in \mathcal{N}$, therefore, the index i can be omitted.*

Corollary 10. *A multi-agent system with a strongly connected network topology $(\mathcal{N}, \mathcal{A})$ employs switching consensus protocol (6.36). Then, $\forall \mathbf{x}(0) \in \mathsf{S}^n$,*

$$\exists k_s \in \mathbb{N}_0 : \ \forall k \geq k_s, \ T(k) \geq \tilde{k}. \tag{6.39}$$

Proof. By construction, $T(k)$ is a non-decreasing unbounded sequence. □

Remark 12. *It is straightforward to come up with a suitable k_s. In fact,*

$$T(k) \geq \tilde{k} \iff \lceil \log_2(k) - 1 \rceil \geq \log_2(\tilde{k}).$$

The latter is true for each iteration $k \in \mathcal{N}$, such that

$$k \geq 2^{1+\log_2(\tilde{k})} = 2\tilde{k} = k_s. \tag{6.40}$$

Lemma 14. *A multi-agent system with a strongly connected network topology $(\mathcal{N}, \mathcal{A})$ employs protocol (6.36). Then, $\forall \mathbf{x}(0) \in \mathsf{S}^n$,*

$$\forall k \geq k_s, \ \exists j \in \mathcal{N} \setminus \mathcal{M}_{T(k)} \implies \mathcal{M}_{T(k)} \subset \mathcal{M}_{2T(k)+2}. \tag{6.41}$$

In words: if max-consensus has not yet been achieved at iteration $T(k)$, the number of non-maximal agents will be strictly smaller at iteration $2T(k)+2$.

Proof. By Proposition 31, given the left-hand side of (6.41), there exists a maximal agent in the neighborhood of a non-maximal agent i, i.e. $\exists i \in \mathcal{N} \setminus \mathcal{M}_{T(k)} : N_i \cap \mathcal{M}_{T(k)} \neq \emptyset$. By (6.36a), for $k = 2T_i(k)$,

$$\forall i \in \mathcal{N}, \ y_i(2T_i(k)+1) = \prod_{t=T_i(k)}^{2T_i(k)} y_i(t).$$

By Corollary 10, $\forall k \geq k_s$ (i.e., $T_i(k) \geq \tilde{k}$), the following holds:

$$\forall i \in \{i \in \mathcal{N} \mid N_i \cap \mathcal{M}_{T(k)} \neq \emptyset\}, \ x_i(2T_i(k)+2) = x^*, \tag{6.42}$$

meaning that agent i will become maximal at instant $2T(k)+2$. □

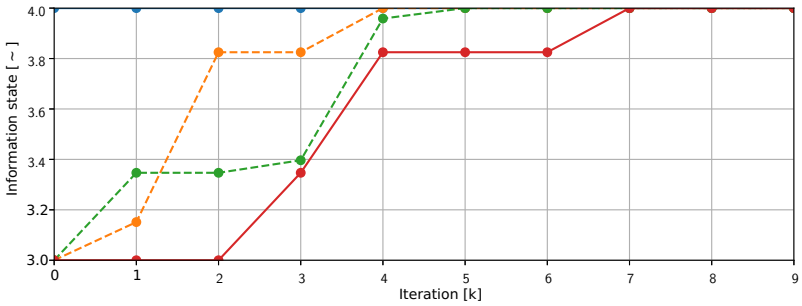


Figure 6.8: Similar to Figure 6.7, but in the presence of a fading channel, the system achieves finite-time max-consensus. However, the evolution through iterations of the information states of agents a and c (dashed lines) are different. This is since, in general, due to the presence of channel coefficients, $u_a(k) \neq u_c(k)$.

Given the above results, it is straightforward to show that finite-time max-consensus is deterministically achieved by employing protocol (6.36).

Theorem 11. *Given a multi-agent system with a strongly connected network topology* $(\mathcal{N}, \mathcal{A})$ *and initial information state* $\mathbf{x}(0) \in \mathbb{S}^n$. *If agents employ the switching consensus protocol* (6.36), *the system achieves finite-time max-consensus.*

Proof. By Lemma 14, $\forall k \geq k_s$, the number of maximal agents strictly increases between $k = T(k)$ and $k = 2T(k) + 2$, unless $\mathcal{N} = \mathcal{M}_{T(k)}$. Therefore, $\forall k \geq k_s$,

$$\mathcal{M}_{T(k)} \subset \mathcal{M}_{2T(k)+2} \subset \cdots \subseteq \mathcal{N}, \tag{6.43}$$

which is equivalent to

$$|\mathcal{M}_{T(k)}| < |M_{2T(k)+2}| < \cdots \leq |\mathcal{N}|. \tag{6.44}$$

As $\mathcal{N}$ is a finite set, it is obvious that this process is finished after a finite numbers of steps. □

6.2.3 *Simulations*

The following numerical experiments illustrates that multi-agent systems with strongly connected network topologies indeed achieve finite-time max-consensus by employing the switching extended consensus protocol (6.36).

Example 5. *The multi-agent system, with network topology* $(\mathcal{N}, \mathcal{A})$ *as in Figure 6.1 and with identical and constant channel coefficients, is endowed with the switching consensus protocol* (6.36) *and the simulation result is shown in Figure 6.7. Unlike Example 1, finite-time max-consensus is achieved. At instant* $k = 2T(0) + 1 = 5$, *all non-maximal agents have lost authorization to broadcast; by this, at instant* $k = 2T(0) + 2 = 6$, *all those agents including a maximal agent in their neighbourhood become maximal as well. In the case of a fading channel, finite-time max-consensus is also achieved, as shown in Figure 6.8.*

Example 6. *In this example, a larger system is analyzed. The number of agents, the network topology, the channel coefficients, and the initial information state are randomly chosen* (*under the only constraint that the network topology has to be strongly connected*), *as shown in Figure 6.9. Such*

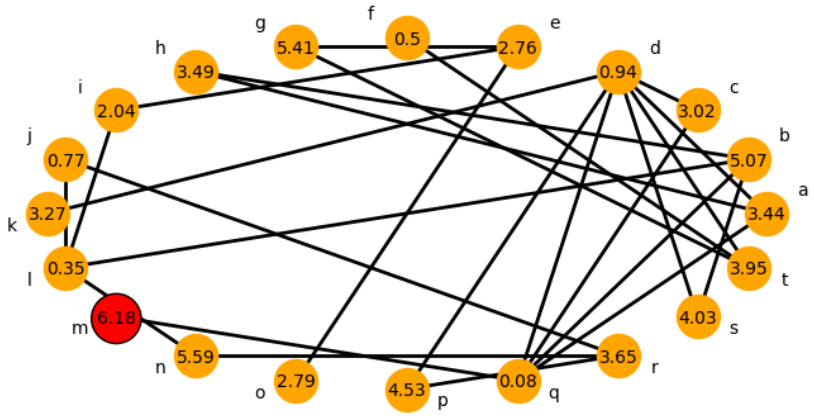


Figure 6.9: Multi-agent system in Example 6 with strongly connected network topology. All arcs are directed, although (for clarity) directions (arrows) are omitted. In fact, we assume that for each arc from node i to node j, there is also one arc from node j to node i. The maximal node is m, and $x_m(0) = 6.18$.

a system, employed with the switching consensus protocol (6.36), achieves finite-time max-consensus, as indicated in Figure 6.10.

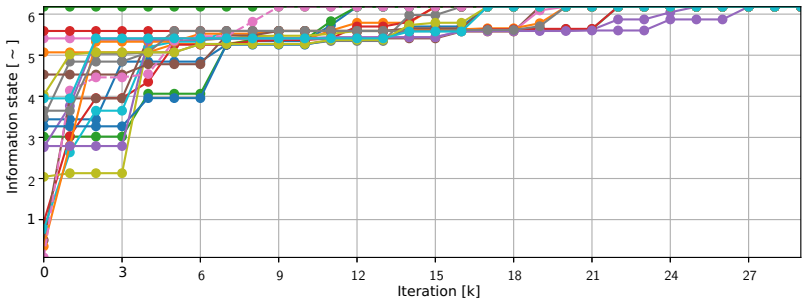


Figure 6.10: Evolution of information states for Example 6. Finite-time max-consensus is achieved in 27 steps.

6.2.4 *Comparison with the standard approach*

In Section 2.3.2, we presented the so-called standard approach. It consists of the combination of an orthogonal channel access communication method and the consensus protocol (2.19). We now compare the standard approach with the extended protocol (6.36), to investigate the benefits of the latter.

In the following, the channel access method used for comparison is TDMA (Time-Division Multiple Access); this method guarantees orthogonal transmissions by dividing each discrete transmission into different time slots. Clearly, each iteration considered in (2.19) then corresponds in reality to n such time slots, since each of the n users has to transmit in a one-after-the-other fashion.

On the other hand, computing inputs for the agents via superposition (cf. (6.3)) takes 2 communication time-slots (see Section 4.1), independently of the network size, in order to obtain the normalized channel fading coefficients. Yet, consensus protocol (6.36) requires, in general, a higher number of iterations than the standard approach, and it depends on channel realization. Therefore, a meaningful comparison can be only done via randomized simulations.

For networks of size between 3 and 100, one hundred different simulations are executed. Each one represents a random initial vector and a random (connected) topology. $\forall k \in \mathbb{N}_0$, $\forall i \in \mathcal{N}$, $\forall j \in N_i(k)$, the random channel coefficients $\xi_{ij}(k)$ are drawn out of independent Rayleigh distributions with variance 1. $\bar{k}_t$ denotes the number of time slots required by the traditional approach for achieving max-consensus, and $\bar{k}_b$ the number of time-slots required by the switching protocol (6.36) to ensure max-consensus. For each experiment, in Figure 6.11, we plot the ratio of the two quantities, defined as $r = \frac{\bar{k}_t}{\bar{k}_b}$. The numerical experiment shows that for multi-agent systems composed of more than approximately 15 agents, employing (6.36) and channel superposition saves significant convergence time.

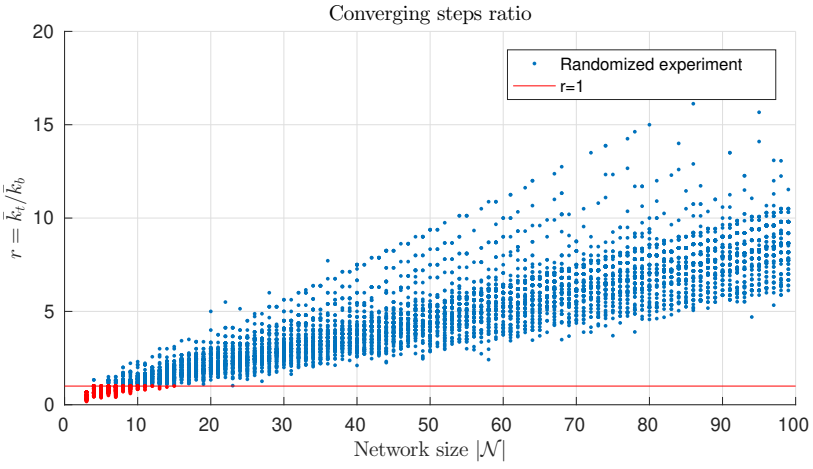


Figure 6.11: Each point comes from a randomized experiment, whose abscissa represents the network size and its whose ordinate is the ratio $\bar{k}_t/\bar{k}_b$. All points above the red line correspond to those cases when the here proposed max-consensus protocol performs better than the traditional approach.

7
Formation Control

Partial results of the presented section have been published in Molinari and Raisch [60]. A special thanks to Dr. Mirko Mauri for the mathematical discussion concerning Lemma 17.

Formation control strategies allow agents moving in space[1] to spread out around a point, referred to as *centroid*, which has to be distributively agreed upon. The position of agent $i \in \mathcal{N}$ at a given time $t \in \mathbb{R}_{\geq 0}$ is in vector $\mathbf{x}_i(t) \in \mathbb{R}^p$, $p \in \mathbb{N}$. The kinematics of each agent $i \in \mathcal{N}$ is given as

$$t \in \mathbb{R}_{\geq 0}, \qquad \dot{\mathbf{x}}_i(t) = f(\mathbf{x}_i(t), \mathbf{u}_i(t)) \tag{7.1}$$

where $\mathbf{u}_i(t) \in \mathbb{R}^m$, $m \leq p$, is the input vector. Depending on the considered dynamics, $\mathbf{x}_i(t)$ can also include orientation and speed. Agents communicate and exchange information with each other. In a realistic framework, communication across the network cannot be modeled as a continuous flow of information; instead, agents transmit and receive data only at discrete update times $t_k \in \mathbb{R}_{\geq 0}$, $k \in \mathbb{N}_0$. Let $\mathcal{T}_k$ be the sequence of update times, namely

$$\mathcal{T}_k := \{t_k\}_{k \in \mathbb{N}_0}. \tag{7.2}$$

In the following, we assume that the interval $\Delta_k := t_{k+1} - t_k$ between two subsequent update times is bounded from below and above, i.e.

$$\exists \underline{\Delta}, \bar{\Delta} \in \mathbb{R}_{\geq 0} : \quad \underline{\Delta} \leq \Delta(k) \leq \bar{\Delta}, \forall k \in \mathbb{N}_0. \tag{7.3}$$

The communication network topology at every update time t_k is modeled by a directed graph $\mathcal{G}(k) = (\mathcal{N}, \mathcal{A}(k))$. Let

$$\mathfrak{G} := \{\mathcal{G}(k)\}_{k \in \mathbb{N}_0} \tag{7.4}$$

be the sequence of graphs. The scope of a formation control problem is to find a distributed control strategy such that the multi-agent system converges to a formation in space, i.e.,

$$\forall i \in \mathcal{N}, \lim_{t \to \infty} \begin{bmatrix} x_i(t) \\ y_i(t) \end{bmatrix} = \mathbf{x}^* + \mathbf{d}_i, \tag{7.5}$$

where $\mathbf{x}^* \in \mathbb{R}^2$ is the so-called *centroid* of the formation and $\mathbf{d}_i \in \mathbb{R}^2$ is the desired displacement of agent $i \in \mathcal{N}$ from the centroid. Desired displacements are given; in fact, we assume that each agent knows a

[1] In what follows, for simplicity of notation and representation, we consider a two-dimensional space.

priori the desired relative position within the formation. This is a realistic assumption, since, most times, each agent has its own role in the swarm, which coincides with a relative position in the final formation. On the other hand, there is no a-priori knowledge about the formation centroid $\mathbf{x}^*$, which has to be distributively negotiated.

7.1 Single-Integrator Dynamics

Partial results of the presented section have been published in Molinari and Raisch [60].

We initially consider agents whose dynamics (along each independent dimension) can be modeled with a single-integrator. Besides a purely didactic purpose, the investigation of singe-integrator dynamics will turn to be useful also for more complex (and realistic) conditions, see Section 7.2.1. In what follows, for ease of representation, we consider $p = 2$, namely a two-dimensional formation control problem.

Formally, the dynamics of each agent $i \in \mathcal{N}$ is

$$\dot{\mathbf{x}}_i(t) = \mathbf{u}_i(t), \tag{7.6}$$

where state and input vector are, respectively,

$$\mathbf{x}_i(t) = \begin{bmatrix} x_i(t) \\ y_i(t) \end{bmatrix} \tag{7.7}$$

and

$$\mathbf{u}_i(t) = \begin{bmatrix} u_i^x(t) \\ u_i^y(t) \end{bmatrix}. \tag{7.8}$$

Movements along the two dimensions are clearly independent. Positions are initialized as,

$$\forall i \in \mathcal{N},\ \mathbf{x}_i(0) = \mathbf{x}_{i_0}.$$

The goal is to find a distributed control strategy such that the multi-agent system converges to a formation in the plane, see (7.5), where both centroid and displacement are two-dimensional quantities, namely, $\mathbf{x}^* \in \mathbb{R}^2$, $\mathbf{d}_i \in \mathbb{R}^2$. Note that $\mathbf{x}^*$ is not known a priori, but has to be distributively negotiated by agents. On the other hand, each agent $i \in \mathcal{N}$ knows its desired relative position in the future formation.

7.1.1 Communication Protocol

At every update time $t_k \in \mathcal{T}_k$, $k \in \mathbb{N}_0$, each agent $j \in \mathcal{N}$ broadcasts three (i.e., $p + 1$) mutually orthogonal signals (in the same way as presented in Section 4), namely,

$$\tau_i^x(t_k) := x_i(t_k) - d_i^x, \tag{7.9}$$

$$\tau_i^y(t_k) := y_i(t_k) - d_i^y, \tag{7.10}$$

and

$$\tau_i'(t_k) := 1. \tag{7.11}$$

Signals $\tau_i^x(t_k)$, respectively $\tau_i^y(t_k)$, can be interpreted as the local estimate at agent $i \in \mathcal{N}$ of the centroid abscissa, respectively ordinate. Signal $\tau_i'(t_k)$ is needed to handle the unknown channel coefficients in

the WMAC model, as already done in Section 5. By the noiseless fading WMAC model (see Section 4.2.2), each agent $i \in \mathcal{N}$ receives three orthogonal signals, i.e.

$$\tilde{\nu}_i^x(t_k) := \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) \tau_j^x(t_k), \tag{7.12}$$

$$\tilde{\nu}_i^y(t_k) := \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) \tau_j^y(t_k), \tag{7.13}$$

and

$$\tilde{\nu}_i'(t_k) := \sum_{j \in \mathcal{N}_i(k)} \xi_{ij}(k). \tag{7.14}$$

Since channel coefficients are positive, and assuming that each agent has at least one neighbor[2]

$$\forall t_k \in \mathcal{T}_k,\ \forall i \in \mathcal{N},\ \tilde{\nu}_i'(t_k) \in \mathbb{R}_{>0}. \tag{7.15}$$

[2] This condition comes directly as a consequence of the underlying topology being strongly connected.

By (7.15) it is possible to normalize the received signals as follows

$$\nu_i^x(t_k) := \frac{\tilde{\nu}_i^x(t_k)}{\tilde{\nu}_i'(t_k)} = \frac{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) \tau_j^x(t_k)}{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)}, \tag{7.16}$$

$$\nu_i^y(t_k) := \frac{\tilde{\nu}_i^y(t_k)}{\tilde{\nu}_i'(t_k)} = \frac{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k) \tau_j^y(t_k)}{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)}. \tag{7.17}$$

In what follows, let, $\forall i, j \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $h_{ij}(k)$ be the normalized fading coefficient for broadcast transmission from j to i at update step k, defined as

$$h_{ij}(k) := \begin{cases} \dfrac{\xi_{ij}(k)}{\sum\limits_{j \in \mathcal{N}_i(k)} \xi_{ij}(k)} & \text{if } (j,i) \in \mathcal{A}(k) \\ 0 & \text{otherwise} \end{cases}. \tag{7.18}$$

Observation 4. *By construction,*

$$\forall i, j \in \mathcal{N},\ \forall k \in \mathbb{N}_0,\ h_{ij}(k) \in [0,1]. \tag{7.19}$$

Moreover, $\forall i \in \mathcal{N},\ \forall k \in \mathbb{N}_0$,

$$\sum_{j=1}^{n} h_{ij}(k) = \sum_{j \in \mathcal{N}_i(k)} h_{ij}(k) = 1. \tag{7.20}$$

By (7.18), $\nu_i^x(t_k)$ and $\nu_i^y(t_k)$ can be rewritten as

$$\nu_i^x(t_k) = \sum_{j=1}^{n} h_{ij}(k) \tau_j^x(t_k), \tag{7.21}$$

$$\nu_i^y(t_k) = \sum_{j=1}^{n} h_{ij}(k) \tau_j^y(t_k). \tag{7.22}$$

By Observation 4, $\nu_i^x(t_k)$ is a convex combination of signals in set $\left\{\tau_j^x(t_k)\right\}_{j \in \mathcal{N}_i(k)}$, thus a convex combination of neighbors' local guesses of centroid position on the x axis. The same holds also for the second dimension.

Proposition 34. *If the formation is achieved at $t_k \in \mathcal{T}_k$ (cf. (7.5)), then*

$$\forall i \in \mathcal{N}, \ \begin{bmatrix} \nu_i^x(t_k) \\ \nu_i^y(t_k) \end{bmatrix} = \mathbf{x}^*. \tag{7.23}$$

Proof. By (7.5), agents are in the desired formation at $t_k \in \mathcal{T}_k$, if and only if

$$\forall i \in \mathcal{N}, \ \mathbf{x}_i(t_k) - \mathbf{d}_i = \mathbf{x}^*.$$

The latter implies that

$$\begin{bmatrix} \tau_i^x(t_k) \\ \tau_i^y(t_k) \end{bmatrix} = \mathbf{x}^*. \tag{7.24}$$

By showing that $[\nu_i^x(t_k), \nu_i^y(t_k)]'$ is a convex combination of the same value $\mathbf{x}^*$, the proof is concluded. □

7.1.2 Consensus-based Formation Control Protocol

In the following, since the dynamics in the x and y coordinates in (7.6) are decoupled, we analyze only the dynamics in x. An equivalent control strategy will be also applied to the y dynamics. The following control strategy is inspired by [2], where consensus is achieved in a network of continuous-time agents with asynchronous discrete-time updates. An extended consensus protocol is proposed here for achieving formation in those contexts where agents exploit the superposition property of the channel. Because we use broadcast transmission, synchronism in transmission and update is required.

We introduce an additional state variable, i.e.

$$\forall i \in \mathcal{N}, \ \forall t \in \mathbb{R}_{\geq 0}, \ \theta_i^x(t) \tag{7.25}$$

that will serve as a reference variable for the position of agent i along the x axis, namely $x_i(t)$ (analogously, $\theta_i^y(t)$ will have the same functionality for the y axis). While $x_i(t)$ must be always continuous (it represent the position), $\theta_i^x(t)$ can have discontinuities at update times. The control strategy will be the combination of continuous-time dynamics (*flow dynamics*) and discrete-time updates (*jump dynamics*). The proposed control strategy is as follows:

- *Jump dynamics*. $\forall t_k \in \mathcal{T}_k$,

$$\begin{cases} x_i(t_k^+) = x_i(t_k) \\ \theta_i^x(t_k^+) = (1 - \sigma(t_k))\theta_i^x(t_k) + \sigma(t_k)d_i^x + \sigma(t_k)\nu_i^x(t_k) \end{cases}, \tag{7.26}$$

- *Flow dynamics*. $\forall t \in (t_k, t_{k+1}]$, $k \in \mathbb{N}_0$,

$$\begin{cases} \dot{x}_i(t) = -a_i^x(t_k)(x_i(t) - \theta_i^x(t)) \\ \dot{\theta}_x^i(t) = b_i^x(t_k)(x_i(t) - \theta_i^x(t)) \end{cases}, \tag{7.27}$$

where $\bullet(t_k^+)$ refers to the value of a signal $\bullet$ immediately after the *jump* at time t_k, $\nu_i^x(t_k)$ as in (7.21), and $\sigma(t_k) \in (0, 1)$, $a_i^x(t_k)$, $b_i^x(t_k) \in \mathbb{R}_{>0}$

are design parameters. Note that, in general, $\sigma(t_k)$, $a_i^x(t_k)$, $b_i^x(t_k)$ are going to be time-invariant.

Between update times, the flow dynamics (7.27) is executed, which reduces the absolute difference between the two state variables. In fact,

$$x_i(t) > \theta_i^x(t) \implies \dot{x}_i(t) < 0 \wedge \dot{\theta}_x^i(t) > 0$$

and

$$x_i(t) < \theta_i^x(t) \implies \dot{x}_i(t) > 0 \wedge \dot{\theta}_x^i(t) < 0.$$

Clearly, the equilibrium for the flow dynamics (7.26) is $x_i(t) = \theta_i^x(t)$.

On the other hand, the jump dynamics (7.26) keeps the positional variable x_i constant, while it updates the reference value θ_x^i. As in Section 5, we can address $\sigma(t_k)$ as the anti-stubbornness parameter; in fact, the closer $\sigma(t_k)$ is to 1, the more agent i relies on the received value $\nu_i^x(t_k)$.

The following theorem shows that agents with dynamics (7.26)-(7.27) achieve the desired formation asymptotically (cf. (7.5)) under a sufficient condition: $\mathfrak{G}$ (defined in (7.4) as the sequence of communication network topologies) is a sequence of strongly connected graphs. Note that the final centroid position $\mathbf{x}^*$ is not known a priori, but is negotiated during the execution.

Theorem 12. *Consider a set of communicating agents with dynamics (7.26)-(7.27). If $\mathfrak{G}$ is a sequence of strongly connected topologies, then the system achieves the desired formation in the sense of (7.5).*

The proof of this theorem requires a number of auxiliary results. Define shift of state coordinates by, $\forall t \in \mathbb{R}_{>0}$,

$$\tilde{x}_i(t) := x_i(t) - d_i^x, \tag{7.28}$$

$$\tilde{\theta}_i^x(t) := \theta_i^x(t) - d_i^x. \tag{7.29}$$

Observation 5. *Consensus in $\tilde{x}_i$ implies (7.5).*

Equations (7.26)-(7.27) can be rewritten in terms of the new state variables, i.e.

- *Jump dynamics.* $\forall t_k \in \mathcal{T}_k$,

$$\begin{cases} \tilde{x}_i(t_k^+) = \tilde{x}_i(t_k) \\ \tilde{\theta}_i^x(t_k^+) = (1 - \sigma(t_k))\tilde{\theta}_i^x(t_k) + \sigma(t_k)\nu_i^x(t_k) \end{cases}, \tag{7.30}$$

- *Flow dynamics.* $\forall t \in (t_k, t_{k+1}]$, $k \in \mathbb{N}_0$,

$$\begin{cases} \dot{\tilde{x}}_i(t) = -a_i^x(t_k)(\tilde{x}_i(t) - \tilde{\theta}_i^x(t)) \\ \dot{\tilde{\theta}}_x^i(t) = b_i^x(t_k)(\tilde{x}_i(t) - \tilde{\theta}_i^x(t)) \end{cases}. \tag{7.31}$$

Also the received value $\nu_i^x(t_k)$ is written in terms of the new state variable $\tilde{x}_i(t)$ as

$$\nu_i^x(t_k) = \sum_{j=1}^{n} h_{ij}(k)\tilde{x}_j(t_k). \tag{7.32}$$

A state-transition matrix $\Phi_i(t_k) \in \mathbb{R}^{2\times 2}$ is used to obtain the general solution of (7.31), i.e.,

$$\forall k \in \mathbb{N}_0, \quad \begin{bmatrix} \tilde{x}_i(t_{k+1}) \\ \tilde{\theta}^i_x(t_{k+1}) \end{bmatrix} = \Phi_i(t_k) \begin{bmatrix} \tilde{x}_i(t_k^+) \\ \tilde{\theta}^i_x(t_k^+) \end{bmatrix}. \tag{7.33}$$

The following proposition characterizes the transition matrix $\Phi_i(t_k)$.

Proposition 35. *The entries of the state transition matrix*

$$\Phi_i(t_k) = \begin{bmatrix} \Phi^a_i(t_k) & \Phi^b_i(t_k) \\ \Phi^c_i(t_k) & \Phi^d_i(t_k) \end{bmatrix} \tag{7.34}$$

are

$$\Phi^a_i(t_k) = \frac{a^x_i(t_k)e^{-(a^x_i(t_k)+b^x_i(t_k))(t_{k+1}-t_k^+)} + b^x_i(t_k)}{a^x_i(t_k) + b^x_i(t_k)}, \tag{7.35}$$

$$\Phi^b_i(t_k) = \frac{a^x_i(t_k)(1 - e^{-(a^x_i(t_k)+b^x_i(t_k))(t_{k+1}-t_k^+)})}{a^x_i(t_k) + b^x_i(t_k)}, \tag{7.36}$$

$$\Phi^c_i(t_k) = \frac{b^x_i(t_k)(1 - e^{-(a^x_i(t_k)+b^x_i(t_k))(t_{k+1}-t_k^+)})}{a^x_i(t_k) + b^x_i(t_k)}, \tag{7.37}$$

$$\Phi^d_i(t_k) = \frac{b^x_i(t_k)e^{-(a^x_i(t_k)+b^x_i(t_k))(t_{k+1}-t_k^+)} + a^x_i(t_k)}{a^x_i(t_k) + b^x_i(t_k)}. \tag{7.38}$$

Proof. Observe that (7.31) is a continuous-time linear system with no inputs. Therefore, it can be rewritten as

$$\begin{bmatrix} \dot{\tilde{x}}_i(t) \\ \dot{\tilde{\theta}}^i_x(t) \end{bmatrix} = A_i(t_k) \begin{bmatrix} \tilde{x}_i(t) \\ \tilde{\theta}^i_x(t) \end{bmatrix},$$

where the dynamics matrix $A_i(t_k)$ is

$$A_i(t_k) = \begin{bmatrix} -a^x_i(t_k) & a^x_i(t_k) \\ b^x_i(t_k) & -b^x_i(t_k) \end{bmatrix}. \tag{7.39}$$

It is straightforward to compute two eigenpairs of $A_i(t_k)$, i.e.

$$(\lambda_1 = 0,\ \mathbf{v}_1 = [1, 1]')$$

and

$$(\lambda_2 = -(a^x_i(t_k) + b^x_i(t_k)),\ \mathbf{v}_2 = [a^x_i(t_k), -b^x_i(t_k)]').$$

By this,

$$\Phi_i(t_k) = [\mathbf{v}_1, \mathbf{v}_2] \begin{bmatrix} e^{\lambda_1(t_{k+1}-t_k^+)} & 0 \\ 0 & e^{\lambda_2(t_{k+1}-t_k^+)} \end{bmatrix} [\mathbf{v}_1, \mathbf{v}_2]^{-1},$$

from which (7.35)-(7.38) follows. □

Observation 6. $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, *matrix* $\Phi_i(t_k)$ *is positive and row-stochastic by construction.*

This is easy to see, as $a^x_i(t_k)$, $b^x_i(t_k)$, and $(t_{k+1} - t_k^+)$ are positive quantities. Thus, $\Phi^a_i(t_k),\ \Phi^b_i(t_k),\ \Phi^c_i(t_k),\ \Phi^d_i(t_k) > 0$. Furthermore, $\Phi^a_i(t_k) + \Phi^b_i(t_k) = \Phi^c_i(t_k) + \Phi^d_i(t_k) = 1$.

Let us now define the overall system state vector in the shifted coordinate system (regarding movements in x direction) as

$$\tilde{\mathbf{x}}(t) = [\tilde{x}_1(t), \dots, \tilde{x}_n(t), \tilde{\theta}_1^x(t), \dots, \tilde{\theta}_n^x(t)]'. \tag{7.40}$$

The state evolution during each interval between t_k^+ and t_{k+1} can be described as

$$\tilde{\mathbf{x}}(t_{k+1}) = \mathbf{\Phi}(t_k)\tilde{\mathbf{x}}(t_k^+) \tag{7.41}$$

where, $\forall k \in \mathbb{N}_0$,

$$\mathbf{\Phi}(t_k) = \begin{bmatrix} \mathbf{\Phi}^a(t_k) & \mathbf{\Phi}^b(t_k) \\ \mathbf{\Phi}^c(t_k) & \mathbf{\Phi}^d(t_k) \end{bmatrix} \tag{7.42}$$

with

$$\mathbf{\Phi}^a(t_k) = \text{diag}\left(\Phi_1^a(t_k) \dots \Phi_n^n(t_k)\right) \tag{7.43}$$
$$\mathbf{\Phi}^b(t_k) = \text{diag}\left(\Phi_1^b(t_k) \dots \Phi_n^b(t_k)\right) \tag{7.44}$$
$$\mathbf{\Phi}^c(t_k) = \text{diag}\left(\Phi_1^c(t_k) \dots \Phi_n^c(t_k)\right) \tag{7.45}$$
$$\mathbf{\Phi}^d(t_k) = \text{diag}\left(\Phi_1^d(t_k) \dots \Phi_n^d(t_k)\right) \tag{7.46}$$

Observation 7. *$\forall k \in \mathbb{N}_0$, matrix $\mathbf{\Phi}(t_k)$ is nonnegative row-stochastic by construction.*

Let's put the flow dynamics aside for now and let's analyze the jump dynamics (7.30), (7.32)). It can be rewritten in compact form as, $\forall k \in \mathbb{N}_0$,

$$\tilde{\mathbf{x}}(t_k^+) = \Psi(t_k)\tilde{\mathbf{x}}(t_k), \tag{7.47}$$

where

$$\Psi(t_k) = \begin{bmatrix} \mathbb{I}_n & \mathbf{0}_n \\ \sigma(t_k)\mathfrak{A}(k) & (1-\sigma(t_k))\mathbb{I}_n \end{bmatrix}. \tag{7.48}$$

Matrix $\mathfrak{A}(k)$ is the adjacency matrix associated with the network topology at iteration $t_k \in \mathcal{T}_k$, namely $\mathcal{G}(k)$, with normalized fading coefficients as weights, i.e.,

$$\forall i \in \mathcal{N},\ \forall j \in \mathcal{N},\ [\mathfrak{A}(k)]_{ij} = h_{ij}(k). \tag{7.49}$$

The following results determines some important properties of the adjacency matrix of a strongly connected graph.

Proposition 36. *Given a strongly connected $\mathcal{G}(k)$, matrix $\mathfrak{A}(k)$ is nonnegative, irreducible, and row-stochastic.*

Proof. According to (7.18), $\forall i, j \in \mathcal{N},\ \forall k \in \mathbb{N}_0, h_{ij}(k) \in [0, 1]$. Following from the definition of adjacency matrix and by having nonnegative weights, $\mathfrak{A}(k)$ is nonnegative. By Observation 4, $\sum_{j=1}^n h_{ij}(k) = 1$, hence $\mathfrak{A}(k)$ is row-stochastic. Finally, by [37, Theorem 6.2.24], the adjacency matrix of a strongly connected graph is irreducible. This concludes the proof. □

As a consequence, the following corollary characterizes some properties of $\Psi(t_k)$.

Corollary 11. *$\forall k \in \mathbb{N}_0$, $\Psi(t_k)$ is a nonnegative row-stochastic matrix.*

Proof. Nonnegativity of $\Psi(t_k)$ is a consequence of the nonnegativity of $\mathfrak{A}(k)$ and the fact that $\sigma(t_k) \in (0,1)$. It is also promptly evident that each of the first n rows sums up to 1. Let now be l any value in $n+1 \dots 2n$. The l-th row sum of $\Psi(t_k)$ is

$$\begin{aligned}\sum_{j=1}^{2n}[\Psi(t_k)]_{lj} &= \sum_{j=1}^{n}[\sigma(t_k)\mathfrak{A}(k)]_{lj} + \sum_{j=1}^{n}[(1-\sigma(t_k))\mathbb{I}_n]_{lj} \\ &= \sigma(t_k))\sum_{j=1}^{n}[\mathfrak{A}(k)]_{lj} + (1-\sigma(t_k)) = 1,\end{aligned}$$

as $\mathfrak{A}(k)$ is row-stochastic. This concludes the proof. □

Proposition 37 ([100, Prop. 1.2])**.** *A $n \times n$ nonnegative matrix*

$$A = \begin{bmatrix} A_1 & B \\ C & \mathbf{0}_{(n-l)\times(n-l)} \end{bmatrix}$$

is given, where A_1 is a $l \times l$ $(1 \leq l \leq n)$ irreducible square matrix. If A contains no zero row or zero column, then A is irreducible.

Proof. See [100, Proof of Prop. 1.2]. □

With this result at hand, it is possible to study the matrix

$$\Omega(t_k) := \mathbf{\Phi}(t_k)\Psi(t_k),$$

which is the dynamics matrix of system (7.30)-(7.31) in compact form, i.e.

$$\forall k \in \mathbb{N}_0,\ \tilde{\mathbf{x}}(t_{k+1}) = \Omega(t_k)\tilde{\mathbf{x}}(t_k). \tag{7.50}$$

Proposition 38. *If $\mathcal{G}(k)$, $k \in \mathbb{N}_0$, is strongly connected, $\Omega(t_k)$ is irreducible and row-stochastic.*

Proof. $\Omega(t_k)$ comes from the product of two nonnegative row-stochastic matrices. By [113], $\Omega(t_k)$ will also be nonnegative and row-stochastic. To prove its irreducibility, note that (7.42) and (7.48) imply

$$\Omega(t_k) = \begin{bmatrix} \mathbf{\Phi}^a(t_k) + \sigma(t_k)\mathbf{\Phi}^b(t_k)\mathfrak{A}(k) & (1-\sigma(t_k))\mathbf{\Phi}^b(t_k) \\ \mathbf{\Phi}^c(t_k) + \sigma(t_k)\mathbf{\Phi}^d(t_k)\mathfrak{A}(k) & (1-\sigma(t_k))\mathbf{\Phi}^d(t_k) \end{bmatrix}. \tag{7.51}$$

The product $C = AB$ of a diagonal matrix A with positive diagonal entries and a nonnegative irreducible matrix B is an irreducible matrix C, since, by [37, pg. 30], $C \sim B$ and the irreducibility of a matrix depends only on its type (see [113, pg. 735])[3]. By [98, Theorem 1], the sum of a nonnegative and an irreducible matrix is an irreducible matrix. From these two considerations, it immediately follows that $\mathbf{\Phi}^a(t_k) + \sigma(t_k)\mathbf{\Phi}^b(t_k)\mathfrak{A}(k)$ is irreducible. Let's now consider the following matrix

[3] See pg. 10 for a definition of *matrix type*.

$$\tilde{\Omega}(t_k) = \begin{bmatrix} \mathbf{\Phi}^a(t_k) + \sigma(t_k)\mathbf{\Phi}^b(t_k)\mathfrak{A}(k) & (1-\sigma(t_k))\mathbf{\Phi}^b(t_k) \\ \mathbf{\Phi}^c(t_k) + \sigma(t_k)\mathbf{\Phi}^d(t_k)\mathfrak{A}(k) & \mathbf{0}_{n\times n} \end{bmatrix}.$$

By Proposition 37, $\tilde{\Omega}(t_k)$ is irreducible. Let now $\bar{\Omega}(t_k)$ be a nonnegative matrix defined as

$$\tilde{\Omega}(t_k) = \begin{bmatrix} \mathbf{0}_{n\times n} & \mathbf{0}_{n\times n} \\ \mathbf{0}_{n\times n} & (1-\sigma(t_k))\mathbf{\Phi}^d(t_k) \end{bmatrix}.$$

Hence,

$$\Omega(t_k) = \bar{\Omega}(t_k) + \tilde{\Omega}(t_k).$$

Therefore, $\Omega(t_k)$ is the sum of an irreducible and a nonnegative matrix; by [98, Theorem 1], $\Omega(t_k)$ is irreducible. □

Corollary 12. *If $\Omega(t_k)$ is irreducible, it is also primitive.*

Proof. The irreducible matrix $\Omega(t_k)$ has a positive diagonal by construction. By [99, Theorem 1.4], any irreducible matrix with positive diagonal is also primitive. This concludes the proof. □

By (7.50), by analyzing the sequence of matrices

$$\{\Omega(t_k)\}_{k\in\mathbb{N}_0},$$

we can determine the convergence properties of the multiagent system.

Proposition 39. *If $\mathfrak{G}$ is a sequence of strongly connected graphs, then system (7.50) achieves consensus, i.e.*

$$\lim_{t\to\infty} \tilde{\mathbf{x}}(t) = \tilde{\mathbf{x}}^*,$$

where $\tilde{\mathbf{x}}^* = x^*\mathbf{1}_{2n}$, $x^* \in \mathbb{R}$.

Proof. By (7.50),

$$\lim_{k\to\infty} \tilde{\mathbf{x}}(t_k) = \lim_{k\to\infty} \Omega(t_{k-1})\Omega(t_{k-2})\dots\Omega(t_1)\Omega(t_0)\tilde{\mathbf{x}}(0). \tag{7.52}$$

By this, the convergence of (7.50) is analyzed by studying the matrix product

$$\lim_{k\to\infty} \Omega(t_{k-1})\Omega(t_{k-2})\dots\Omega(t_1)\Omega(t_0). \tag{7.53}$$

An important result about the convergence of a sequence of primitive row-stochastic matrices has been presented by Wolfowitz in [113]. Most consensus literature, see, e.g., [92], have been making use of Wolfowitz's theorem for addressing convergence to an agreement in case of a time-varying consensus protocol. In fact, an infinite sequence of primitive row-stochastic square matrices of dimension $2n$ converges to

$$\lim_{t\to\infty} \Omega(t_{k-1})\Omega(t_{k-2})\dots\Omega(t_1)\Omega(t_0) = \mathbf{1}_{2n}\tilde{\boldsymbol{v}}_n', \tag{7.54}$$

where $\tilde{\boldsymbol{v}}_n \in \mathbb{R}^{2n}_{>0}$ and $\mathbf{1}_{2n}'\tilde{\boldsymbol{v}}_n = 1$. Since $\tilde{\mathbf{x}}(t)$ converges and 7.3 gives an upper-bound to $\Delta(k)$,

$$\lim_{t\to\infty} \tilde{\mathbf{x}}(t) = \lim_{k\to\infty} \tilde{\mathbf{x}}(t_k). \tag{7.55}$$

By combining (7.55), (7.52) and (7.54),

$$\lim_{t\to\infty} \tilde{\mathbf{x}}(t) = \lim_{t\to\infty} \tilde{\mathbf{x}}(t_k) = \mathbf{1}_{2n}\tilde{\boldsymbol{v}}_n'\tilde{\mathbf{x}}(0) = x^*\mathbf{1}_{2n}, \tag{7.56}$$

where $x^* = \tilde{\boldsymbol{v}}_n'\tilde{\mathbf{x}}(0)$. □

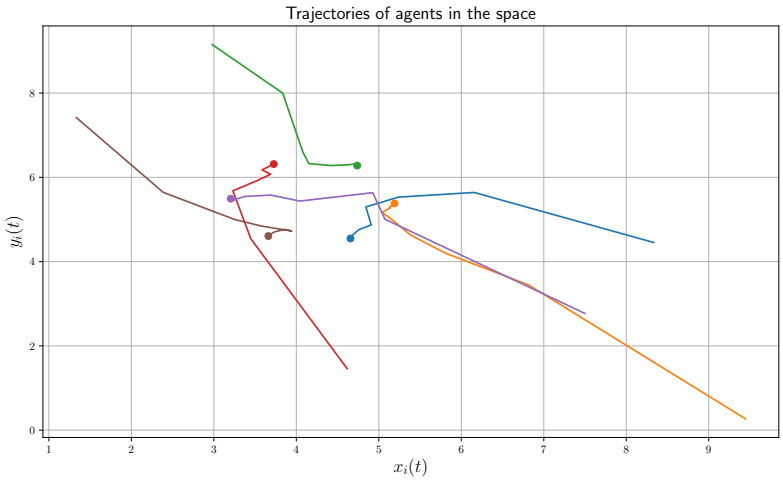


Figure 7.1: Trajectories in the space of agent in $\mathcal{N}$ seeking for a formation. The final achieved shape is a hexagon.

In light of Proposition 39, it is finally possible to formalize the proof for Theorem 12.

Proof of Theorem 12. By Proposition 39, if $\mathfrak{G}$ is a sequence of strongly connected graphs,

$$\forall \tilde{\mathbf{x}}(0) \in \mathbb{R}^{2n}, \ \lim_{t\to\infty} \tilde{\mathbf{x}}(t) = \tilde{\mathbf{x}}^* = x^* \mathbf{1}_{2n}. \tag{7.57}$$

This latter, by (7.28), implies that, $\forall i \in \mathcal{N}$,

$$\lim_{t\to\infty} x_i(t) = x^* + d_i^x. \tag{7.58}$$

As the dynamics in x- and y-directions are decoupled, as discussed on page 121, the analogous result holds for the y-coordinate, i.e., $\forall i \in \mathcal{N}$,

$$\lim_{t\to\infty} y_i(t) = y^* + d_i^y. \tag{7.59}$$

This concludes the proof of Theorem 12. □

7.1.3 *Simulation*

A set $\mathcal{N}$ composed of $n = 6$ agents is given, where, $\forall i \in \mathcal{N}$, $x_i(0)$ and $y_i(0)$ are randomly chosen. The following parameters are used in the simulation, $\forall i \in \mathcal{N}$, $\forall k \in \mathbb{N}_0$, $a_i^x(t_k) = a_i^y(t_k) = 0.5$, $b_i^x(t_k) = b_i^y(t_k) = 0.5$, $\sigma_i(t_k) = 0.8$. Desired displacements from the formation centroid (d_i^x and d_i^y) are given, so that the final desired formation is a hexagon. A sequence of different strongly connected network topologies is randomly chosen. At every update step t_k, $k \in \mathbb{N}_0$, also channel fading coefficients are randomly generated, so that, $\forall i \in \mathcal{N}$, $\forall j \in \mathcal{N}_i(k)$, the coefficients are independent and identically distributed, $\xi_{ij}(k) \sim \mathcal{U}(0,1)$. Also, the sequence $\{\Delta(k)\}_{k\in\mathbb{N}_0}$ is randomly chosen, i.e., $\forall k \in \mathbb{N}_0$, $\Delta(k) \sim \mathcal{U}(10,30)$.
Finally, simulation is run. For solving the differential equations in (7.27), the *odeint* function from Python is used. In Figure 7.1, the two-dimensional trajectories of agents are plotted. Clearly, they converge to the desired hexagonal formation.

7.2 Nonholonomic Dynamics

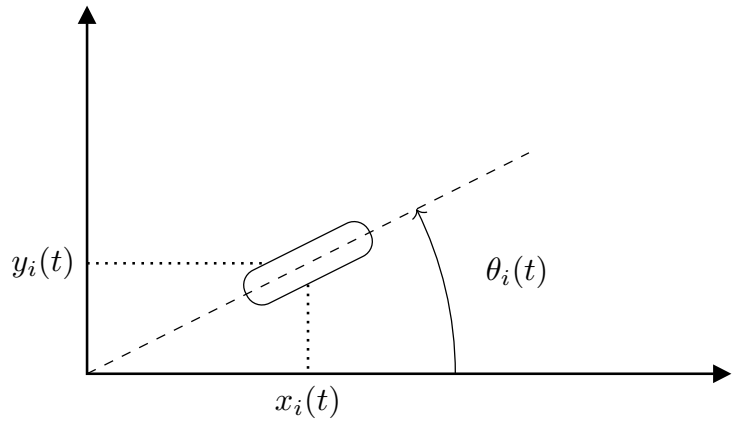


Figure 7.2: Nonholonomic dynamics.

Wheeled Mobile Robots are commonly employed in research and industry when autonomous motion capabilities on the two-dimensional plane are needed. Their dynamics cannot be accurately described by (7.6). In fact, movements along each axis are interdependent. Let the state vector describing the position (and orientation) of each agent $i \in \mathcal{N}$ be,

$$\forall t \in \mathbb{R}_{\geq 0}, \quad \boldsymbol{\xi}_i(t) = \begin{bmatrix} \mathbf{x}_i(t) \\ \theta_i(t) \end{bmatrix} = \begin{bmatrix} x_i(t) \\ y_i(t) \\ \theta_i(t) \end{bmatrix}, \tag{7.60}$$

where $x_i(t)$ (x coordinate of the center of mass), $y_i(t)$ (y coordinate of the center of mass), and $\theta_i(t)$ (angle with regards to the x axis) are as in Figure 7.2. The dynamics of each agent can be modeled by the so-called unicycle dynamics, i.e.,

$$\forall t \in \mathbb{R}_{\geq 0}, \quad \begin{cases} \dot{x}_i(t) &= v_i(t) \cos \theta_i(t) \\ \dot{y}_i(t) &= v_i(t) \sin \theta_i(t) \\ \dot{\theta}_i(t) &= \omega_i(t) \end{cases}, \tag{7.61}$$

where $v_i(t)$ and $\omega_i(t)$ are, respectively, the longitudinal speed and the angular speed of agent $i \in \mathcal{N}$ at time $t \in \mathbb{R}_{\geq 0}$ (and the inputs to the model). System (7.61) is nonholonomic, namely, it has some motion constraints that cannot be expressed in the form

$$f(\boldsymbol{\xi}_i(t), t) = 0.$$

In fact, one constraint is

$$\dot{y}_i(t) = \dot{x}_i(t) \tan \theta_i(t).$$

A thorough analysis of nonholonomic systems and how their motion can be controlled is given in [24]. Although (7.61) is open loop controllable, it has been proven to be nonstabilizable via pure smooth feedback, see [15, 95]. After defining a closed-loop control approach

for the nonholonomic system, a consensus-based formation control strategy can be designed.

The condition for asymptotically achieving a formation, i.e. (7.5), considers only a positional formation, ignoring the orientation (as also done in [14]). In case of nonholonomic robots, in fact, an agreement on the orientation θ_i can be chased once the positional formation is achieved, since, for $v_i(t) = 0$, it is possible to control $\omega_i(t)$ without affecting agents' positions.

7.2.1 *Feedback Linearization Technique*

The first approach aims at linearizing the dynamics, with the same strategy already presented in [9, 82]. By changing variables, it is possible to make the system display a linear dynamics. Accordingly, let P_i be a point placed at a distance $\epsilon \in \mathbb{R}_{>0}$ from the center of mass of robot $i \in \mathcal{N}$ in the direction of the longitudinal velocity, see Figure 7.3.

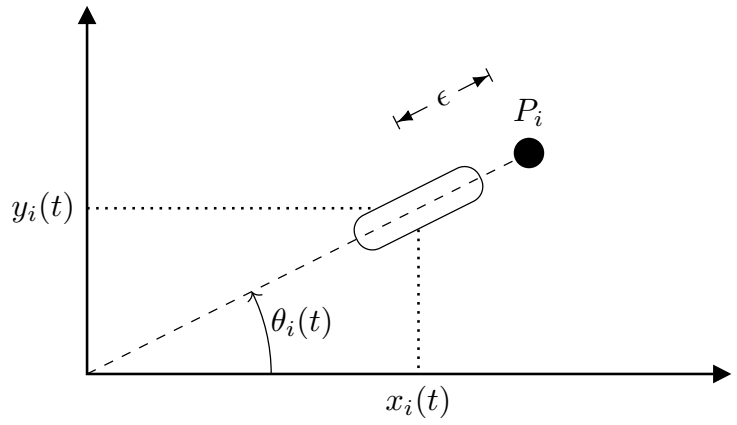


Figure 7.3: Positioning of point P_i.

Its coordinates are

$$\begin{cases} x_{P_i}(t) = x_i(t) + \epsilon \cos\theta_i(t) \\ y_{P_i}(t) = y_i(t) + \epsilon \sin\theta_i(t) \end{cases} . \tag{7.62}$$

This can be differentiated with respect to time, and, by incorporating (7.61), we can control the linear system

$$\begin{cases} \dot{x}_{P_i}(t) = v_{ix}(t) \\ \dot{y}_{P_i}(t) = v_{iy}(t) \end{cases} , \tag{7.63}$$

provided that we satisfy the following equality

$$\begin{bmatrix} v_i(t) \\ \omega_i(t) \end{bmatrix} = \begin{bmatrix} \cos\theta_i(t) & \sin\theta_i(t) \\ -\dfrac{1}{\epsilon}\sin\theta_i(t) & \dfrac{1}{\epsilon}\cos\theta_i(t) \end{bmatrix} \begin{bmatrix} v_{ix}(t) \\ v_{iy}(t) \end{bmatrix} . \tag{7.64}$$

(7.63) is equivalent to (7.6), for which we have already proposed an interference-base solution. It is clear that the system composed of points $[x_{P_i}(t), y_{P_i}(t)]'$ can achieve a formation by exploiting the interference property of the wireless channel by employing control strategy (7.26)-(7.27). The way the nonholonomic dynamics evolves with this control strategy can be investigated by inserting into (7.61) the input vector $[v_i(t), \omega_i(t)]'$ computed in (7.64).
Such an approach makes sense for $\epsilon > 0$. In correspondence of small values of ϵ, we might have high values of $\omega_i(t)$, thus yielding clear implementation issues. By (7.63), each agent $i \in \mathcal{N}$ will control the position of its respective point P_i (rather than its center of mass). Therefore, in a formation control problem of agents with dynamics (7.63),

points P_i will be organized in a formation, rather than agents' centres of mass. Therefore, there will be a noticeable trade-off between accuracy and implementability, as follows:

- small ϵ: P_i is a good approximation for the center of mass of each agent; however, high angular speeds are demanded.
- large ϵ: there is no issue for the saturation of the angular speeds; on the other hand, centers of mass will not be in an accurate formation.

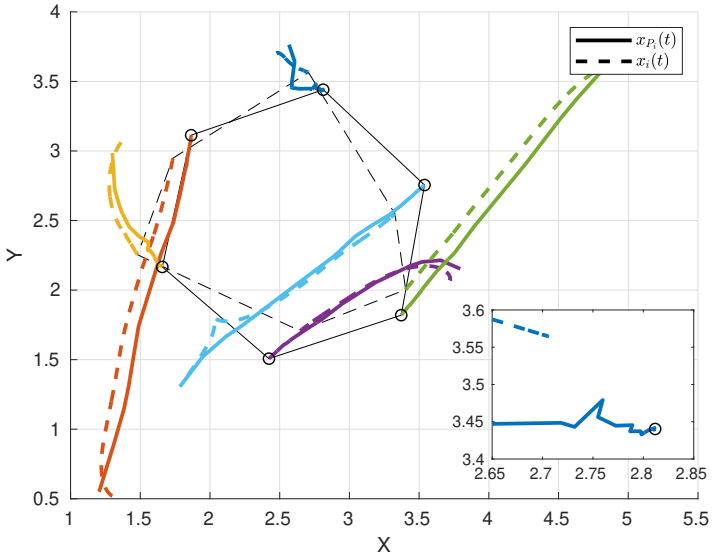


Figure 7.4: Trajectories in the space of agents in $\mathcal{N}$ seeking for a formation with $\epsilon = 0.2$. The final achieved shape of P_i, $i \in \mathcal{N}$, is a hexagon. The same does not apply to the respective centre of mass.

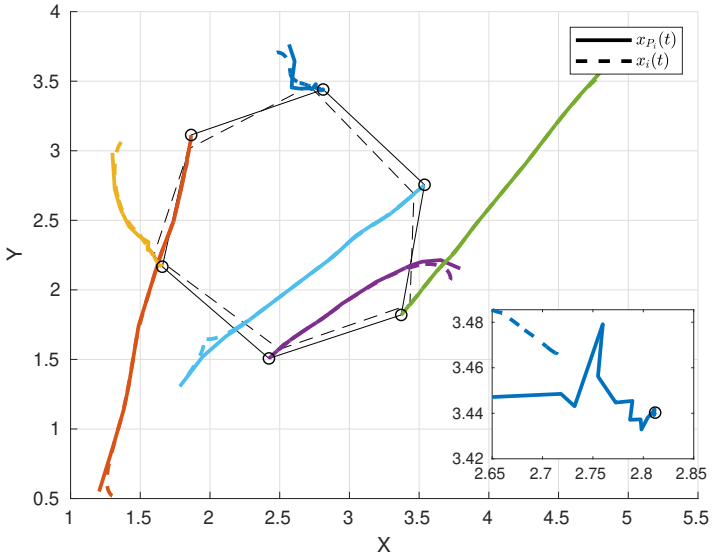


Figure 7.5: Trajectories in the space of agents in $\mathcal{N}$ seeking for a formation. The final achieved shape of P_i, $i \in \mathcal{N}$, is a hexagon. With $\epsilon = 0.1$, centre of mass are also getting closer to a hexagonal formation.

Summarizing, the control approach (here presented for the abscissa, but equivalently also for the ordinate) is as follows:

- *Jump dynamics.* $\forall t_k \in \mathcal{T}_k$,

$$\begin{cases} x_{P_i}(t_k^+) = x_{P_i}(t_k) \\ \theta^x_{P_i}(t_k^+) = (1-\sigma(t_k))\theta^x_{P_i}(t_k) + \sigma(t_k)d_i^x + \sigma(t_k)\nu^x_{P_i}(t_k) \end{cases}, \quad (7.65)$$

- *Flow dynamics.* $\forall t \in (t_k, t_{k+1}]$, $k \in \mathbb{N}_0$,

$$\begin{cases} \dot{x}_{P_i}(t) = -a_i^x(t_k)(x_{P_i}(t) - \theta^x_{P_i}(t)) \\ \dot{\theta}^x_{P_i}(t) = b_i^x(t_k)(x_{P_i}(t) - \theta^x_{P_i}(t)) \end{cases}, \quad (7.66)$$

where all quantities are analogous to the ones already introduced in the previous chapter, but referred to point P_i, rather than to the centre of mass of agent i. Also the quantity $\nu^x_{P_i}(t_k)$ is obtained by exploiting the interference property of the wireless channel, as done in the previous section. The nonholonomic dynamics evolves as (7.61), where inputs are computed as in (7.64), i.e.,

$$\begin{bmatrix} v_i(t) \\ \omega_i(t) \end{bmatrix} = \begin{bmatrix} \cos\theta_i(t) & \sin\theta_i(t) \\ -\frac{1}{\epsilon}\sin\theta_i(t) & \frac{1}{\epsilon}\cos\theta_i(t) \end{bmatrix} \begin{bmatrix} \dot{x}_{P_i}(t) \\ \dot{y}_{P_i}(t) \end{bmatrix}. \quad (7.67)$$

The implementation of (7.26)-(7.27) in this context is experimentally studied in the next section.

7.2.1.1 Simulations

We run numerical simulations in which a network of $n = 6$ agents are trying to achieve a formation by employing the designed strategy. They aim at distributing around a point assuming a hexagonal shape. We first consider the case of $\epsilon = 0.2$. Simulation results are shown in Figure 7.4. As theoretically anticipated, formation of points P_i, $i \in \mathcal{N}$, is achieved. However, robots' centre of masses are not aligned. In Figure 7.4, centre of mass do not converge to a hexagon. By reducing ϵ to a value $\epsilon = 0.1$, as in Figure 7.5, the result obtained in terms of alignment of centre of mass is noticeably better. However, as in theoretical analysis, being ϵ at the denominator in (7.67), such a trajectory requires large values of $\omega_i(t)$. Under a practical point of view, $\omega_i(t)$ cannot be arbitrarily large. The need for a control technique acting on centre of mass motivates the next section.

7.2.2 *Nonlinear Control Technique*

As done in [14] in order to let dynamics (7.61) track a reference trajectory, we implement the piece-wise continuous nonlinear control approach in [25]. One interesting property of this controller is that it does not require infinitely fast switching, as instead sliding controllers do. Our consensus-based formation control strategy is different than what shown in [14]. In fact, in order to exploit the superposition property of the wireless channel, it needs a substantial rephrase of the jump dynamics. Moreover, unlike [14], we are dealing with an inherently directed underlying network topology.

In what follows, we initially present the piece-wise continuous nonlinear control approach presented by [25]. Following, the consensus-based formation control is presented. It has flow and jump dynamics, to cope with the continuous-time nature of (7.61) and the presence of discrete-time broadcast updates. A proof for the convergence to a formation is finally given, together with simulation results.

7.2.2.1 *Piece-wise continuous nonlinear controller*

This Section examines the controller in [25], which considers the exponential stabilization of (7.61) around the origin. Let the compact form of (7.61) be

$$\forall t \in \mathbb{R}_{\geq 0}, \quad \dot{\boldsymbol{\xi}}_i(t) = \psi(\boldsymbol{\xi}_i(t))\mathbf{u}_i(t), \tag{7.68}$$

where

$$\psi(\boldsymbol{\xi}_i(t)) := \begin{bmatrix} \cos\theta_i(t) & 0 \\ \sin\theta_i(t) & 0 \\ 0 & 1 \end{bmatrix}$$

and the vector of inputs is

$$\mathbf{u}_i(t) := \begin{bmatrix} v_i(t) \\ \omega_i(t) \end{bmatrix}.$$

A closed-loop controller

$$\mathbf{u}_i(t) = \kappa(\boldsymbol{\xi}_i(t))$$

is designed as in [25] and is summarized in Appendix A.4, with $v_i(t)$ and $\omega_i(t)$ computed as in (A.46)-(A.47). The following Theorem holds.

Theorem 13 (Theorem 2, [25])**.** *For any initial condition $\boldsymbol{\xi}_i(0) \in \mathbb{R}^3$, the solutions $\boldsymbol{\xi}_i(t),\ t \in \mathbb{R}_{\geq 0}$, of the closed loop equations*

$$\dot{\boldsymbol{\xi}}_i(t) = \psi(\boldsymbol{\xi}_i(t))\kappa(\boldsymbol{\xi}_i(t)), \tag{7.69}$$

converge exponentially to any of the elements in

$$\{(0, 0, 2\pi n),\ n = 0, \pm 1, \pm 2, \dots\}.$$

Note that vector $\boldsymbol{\xi}_i(t)$ includes orientation, whilst vector

$$\mathbf{x}_i(t) = \begin{bmatrix} x_i(t) \\ y_i(t) \end{bmatrix}$$

includes only positions. The following result proves exponential convergence also for the vector of positions itself for system (7.68).

Proposition 40. *For any initial condition* $\mathbf{x}_i(t_0) \in \mathbb{R}^2$, $\exists \varepsilon_i \in \mathbb{R}_{>0}$, $\exists \rho_i \in \mathbb{R}_{>0}$:

$$\forall t > t_0, \qquad \|\mathbf{x}_i(t)\| \leq \varepsilon_i \, \|\mathbf{x}_i(t_0)\| \, e^{-\rho_i (t - t_0)}. \tag{7.70}$$

Proof. The proof is obtained by bringing together [25, Lemma 1] and [25, Theorem 1]. □

Consider now the shift of coordinates as in (7.28), i.e., $\forall t \in \mathbb{R}_{>0}$,

$$\tilde{x}_i(t) := x_i(t) - d_i^x, \tag{7.71}$$

$$\tilde{y}_i(t) := y_i(t) - d_i^y. \tag{7.72}$$

Let $\tilde{\boldsymbol{\xi}}_i(t)$ be, $\forall t \in \mathbb{R}_{\geq 0}$,

$$\tilde{\boldsymbol{\xi}}_i(t) := \boldsymbol{\xi}_i(t) - \begin{bmatrix} d_i^x \\ d_i^y \\ 0 \end{bmatrix}.$$

As in (7.68),

$$\forall t \in \mathbb{R}_{\geq 0}, \quad \dot{\tilde{\boldsymbol{\xi}}}_i(t) = \psi(\tilde{\boldsymbol{\xi}}_i(t)) \mathbf{u}_i(t), \tag{7.73}$$

and, trivially, $\dot{\tilde{\boldsymbol{\xi}}}_i(t) = \dot{\boldsymbol{\xi}}_i(t)$. By Theorem 13, for any initial condition, the solution of the closed-loop equations

$$\dot{\tilde{\boldsymbol{\xi}}}_i(t) = \psi(\tilde{\boldsymbol{\xi}}_i(t)) \kappa(\tilde{\boldsymbol{\xi}}_i(t)) \tag{7.74}$$

converges exponentially to any of the elements in

$$\{(0, 0, 2\pi n)', \; n = 0, \pm 1, \pm 2, \dots\}.$$

By this latter, with the closed-loop dynamics (7.74),

$$\boldsymbol{\xi}_i(t) \to \{(d_i^x, d_i^y, 2\pi n)', \; n = 0, \pm 1, \pm 2, \dots\},$$

and, by Proposition 40,

$$\forall t > t_0, \qquad \left\| \mathbf{x}_i(t) - \begin{bmatrix} d_i^x \\ d_i^y \end{bmatrix} \right\| \leq \varepsilon_i \left\| \mathbf{x}_i(t_0) - \begin{bmatrix} d_i^x \\ d_i^y \end{bmatrix} \right\| e^{-\rho_i (t - t_0)}. \tag{7.75}$$

7.2.2.2 *Consensus-based Formation Control Protocol*

Similarly to what done for the single-integrator case, we define a state variable to be considered as the reference (for the position along x and y axis), namely,

$$\forall i \in \mathcal{N}, \; \forall t \in \mathbb{R}_{\geq 0}, \; \mathbf{r}_i(t) \in \mathbb{R}^2.$$

The reference for the orientation is arbitrarily 0. In fact, as motivated at the beginning of Section 7.2, we are interested only in achieving a formation for the positions (and not for the orientations). We employ

the same communication structure, designed for exploiting interference, as in Section 7.1.1. Accordingly, at every update time $t_k \in \mathcal{T}_k$, each agent receives as input

$$\boldsymbol{\nu}_i(t_k) := \begin{bmatrix} \nu_i^x(t_k) \\ \nu_i^y(t_k) \end{bmatrix} = \sum_{j \in \mathcal{N}_i(k)} h_{ij}(k) \left(\mathbf{x}_j(t_k) - \mathbf{d}_j \right). \tag{7.76}$$

The proposed formation control strategy is as follows:

- *Jump dynamics.* $\forall t_k \in \mathcal{T}_k$,

$$\begin{cases} \mathbf{x}_i(t_k^+) = \mathbf{x}_i(t_k) \\ \theta_i(t_k^+) = \theta_i(t_k) \\ \mathbf{r}_i(t_k^+) = (1 - \sigma(t_k))\mathbf{x}_i(t_k) + \sigma(t_k)\mathbf{d}_i + \sigma(t_k)\boldsymbol{\nu}_i(t_k) \end{cases}, \tag{7.77}$$

- *Flow dynamics.* $\forall t \in (t_k, t_{k+1}]$, $k \in \mathbb{N}_0$,

$$\begin{cases} \begin{bmatrix} \dot{\mathbf{x}}_i(t) \\ \dot{\theta}_i(t) \end{bmatrix} & = \psi \left(\begin{bmatrix} \mathbf{x}_i(t) \\ \theta_i(t) \end{bmatrix} - \begin{bmatrix} \mathbf{r}_i(t) \\ 0 \end{bmatrix} \right) \kappa \left(\begin{bmatrix} \mathbf{x}_i(t) \\ \theta_i(t) \end{bmatrix} - \begin{bmatrix} \mathbf{r}_i(t)] \\ 0 \end{bmatrix} \right). \\ \dot{\mathbf{r}}_i(t) & = \mathbf{0}_2 \end{cases} \tag{7.78}$$

Theorem 14. *Consider a set of communicating agents with dynamics (7.77)-(7.78) and communication system (7.76). If $\mathfrak{G}$ is a sequence of strongly connected topologies,*

$$\exists \underline{\Delta} \in \mathbb{R}_{\geq 0} : \ \forall k \in \mathbb{N}_0, \ \underline{\Delta} \leq \Delta_k,$$

such that the system achieves the desired formation in the sense of (7.5).

7.2.2.3 Results leading to Theorem 14

Consider the following state variables in $\mathbb{R}^2$,

$$\forall i \in \mathcal{N}, \ \forall t \in \mathbb{R}_{\geq 0}, \qquad \mathbf{q}_i(t) := \mathbf{x}_i(t) - \mathbf{r}_i(t), \tag{7.79}$$

$$\mathbf{s}_i(t) := \mathbf{r}_i(t) - \mathbf{d}_i. \tag{7.80}$$

With these new state variables at hand, we can rewrite (7.77)-(7.78), thus obtaining the following dynamics, $\forall i \in \mathcal{N}$:

- *Jump dynamics.* $\forall t_k \in \mathcal{T}_k$,

$$\begin{cases} \mathbf{q}_i(t_k^+) & = \sigma(t_k)(\mathbf{q}_i(t_k) + \mathbf{s}_i(t_k)) - \\ & \qquad \sigma(t_k) \sum_{j \in \mathcal{N}_i(k)} h_{ij}(k)(\mathbf{q}_j(t_k) + \mathbf{s}_j(t_k)) \\ \theta_i(t_k^+) & = \theta_i(t_k) \\ \mathbf{s}_i(t_k^+) & = (1 - \sigma(t_k))(\mathbf{q}_i(t_k) + \mathbf{s}_i(t_k)) + \\ & \qquad \sigma(t_k) \sum_{j \in \mathcal{N}_i(k)} h_{ij}(k)(\mathbf{q}_j(t_k) + \mathbf{s}_j(t_k)) \end{cases}, \tag{7.81}$$

- *Flow dynamics.* $\forall t \in (t_k, t_{k+1}]$, $k \in \mathbb{N}_0$,

$$\begin{cases} \begin{bmatrix} \dot{\mathbf{q}}_i(t) \\ \dot{\theta}_i(t) \end{bmatrix} = \psi \left(\begin{bmatrix} \mathbf{q}_i(t) \\ \theta_i(t) \end{bmatrix} \right) \kappa \left(\begin{bmatrix} \mathbf{q}_i(t) \\ \theta_i(t) \end{bmatrix} \right) \\ \dot{\mathbf{s}}_i(t) = \mathbf{0}_2 \end{cases}, \tag{7.82}$$

The following state variables stack all $\mathbf{q}_i(t)$ and $\mathbf{s}_i(t)$, respectively, i.e.,

$$\forall t \in \mathbb{R}_{\geq 0}, \qquad \mathbf{q}(t) := [\mathbf{q}_1(t)', \ldots, \mathbf{q}_n(t)']' , \tag{7.83}$$

$$\mathbf{s}(t) := [\mathbf{s}_1(t)', \ldots, \mathbf{s}_n(t)']' . \tag{7.84}$$

Jump dynamics (7.81) in compact form, omitting agents' orientation variables, becomes

$$\begin{bmatrix} \mathbf{s}(t_k^+) \\ \mathbf{q}(t_k^+) \end{bmatrix} = \begin{bmatrix} D_n^\sigma(k) \otimes \mathbb{I}_2 & D_n^\sigma(k) \otimes \mathbb{I}_2 \\ (\mathbb{I}_n - D_n^\sigma(k)) \otimes \mathbb{I}_2 & (\mathbb{I}_n - D_n^\sigma(k)) \otimes \mathbb{I}_2 \end{bmatrix} \begin{bmatrix} \mathbf{s}(t_k) \\ \mathbf{q}(t_k) \end{bmatrix}, \tag{7.85}$$

where $D_n^\sigma(k)$ is defined as in (5.14), i.e.,

$$[D_n^\sigma(k)]_{ij} = \begin{cases} (1 - \sigma(t_k)) & \text{if } i = j \\ \sigma(t_k) h_{ij}(k) & \text{if } i \neq j \end{cases},$$

and normalized channel coefficients defined in (5.8). Asymptotic convergence to the desired formation means that

$$\lim_{t \to \infty} \begin{bmatrix} \mathbf{s}(t) \\ \mathbf{q}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{1}_n \otimes \mathbf{x}^* \\ \mathbf{0}_{2n} \end{bmatrix}. \tag{7.86}$$

Let $\mathcal{U} \in \mathbb{R}^{2(n-1) \times 2n}$ be a transformation matrix, already employed in Section 5.1.3.2, such that, in case of asymptotic convergence to the desired formation,

$$\lim_{t \to \infty} \mathcal{U}\mathbf{s}(t) = \mathbf{0}_{2(n-1)}. \tag{7.87}$$

This strategy has been first introduced by [40] and employed by [14]. The characterization of matrix $\mathcal{U}$ can be found in Appendix A.2, which also extends the analysis presented in both [14, 40]. The main properties of $\mathcal{U}$ are here summarized:

- $\mathcal{U}$ is full row-rank;
- $\mathcal{U}(\mathbf{1}_n \otimes a) = \mathbf{0}_{2(n-1)}$,[4] for any $a \in \mathbb{R}^2$.
- $\mathcal{U}\mathcal{U}' = \mathbb{I}_{2(n-1) \times 2(n-1)}$, i.e., rows of $\mathcal{U}$ are orthogonal.

[4] This follows directly from the properties of the Kronecker product and from the definition of U, namely,

$$\begin{aligned} \mathcal{U}(\mathbf{1}_n \otimes a) &= (U \otimes \mathbb{I}_2)(\mathbf{1}_n \otimes a) \\ &= (U\mathbf{1}_n) \otimes a = \mathbf{0}_{2(n-1)}. \end{aligned}$$

Let, $\forall t \in \mathbb{R}_{\geq 0}$,

$$\mathbf{s}_p(t) := \mathcal{U}\mathbf{s}(t). \tag{7.88}$$

Clearly, $\mathbf{s}_p(t) \in \mathbb{R}^{2(n-1)}$. By (7.86) and (7.87), $\mathbf{s}_p(t)$ tends to $\mathbf{0}_{2(n-1)}$ as the system converges to a formation. Thus, (7.86) is equivalent to

$$\lim_{t \to \infty} \begin{bmatrix} \mathbf{s}_p(t) \\ \mathbf{q}(t) \end{bmatrix} = \mathbf{0}_{2(n-1)+2n} \tag{7.89}$$

We can, then, employ as state variable the vector $[\mathbf{s}_p(t)', \mathbf{q}(t)']'$. By multiplying by $\mathcal{U}$ both sides of the first equation in (7.85), one gets

$$\mathcal{U}\mathbf{s}(t_{k+1}) = \mathcal{U}\left(D_n^\sigma(k) \otimes \mathbb{I}_2\right)\mathbf{s}(t_k) + \mathcal{U}\left(D_n^\sigma(k) \otimes \mathbb{I}_2\right)\mathbf{q}(t_k). \tag{7.90}$$

By exploiting the Kronecker product for $\mathcal{U} = U \otimes \mathbb{I}_2$ (see A.14),

$$\mathcal{U}\mathbf{s}(t_{k+1}) = ((UD_n^\sigma(k)) \otimes (\mathbb{I}_2))\,\mathbf{s}(t_k) + ((UD_n^\sigma(k)) \otimes (\mathbb{I}_2))\,\mathbf{q}(t_k). \tag{7.91}$$

By Proposition A.2.2, $UD_n^\sigma(k) = Q^\sigma(k)U$, with $Q^\sigma(k) \in \mathbb{R}^{2(n-1)\times 2(n-1)}$ uniquely defined. Thus,

$$\mathcal{U}\mathbf{s}(t_{k+1}) = ((Q^\sigma(k)U) \otimes (\mathbb{I}_2))\,\mathbf{s}(t_k) + ((Q^\sigma(k)U) \otimes (\mathbb{I}_2))\,\mathbf{q}(t_k). \tag{7.92}$$

By the Kronecker product,

$$\mathcal{U}\mathbf{s}(t_{k+1}) = (Q^\sigma(k) \otimes \mathbb{I}_2)\mathcal{U}\mathbf{s}(t_k) + ((Q^\sigma(k)U) \otimes (\mathbb{I}_2))\,\mathbf{q}(t_k). \tag{7.93}$$

By (7.88),

$$\mathbf{s}_p(t_{k+1}) = (Q^\sigma(k) \otimes \mathbb{I}_2)\,\mathbf{s}_p(t_k) + ((Q^\sigma(k)U) \otimes (\mathbb{I}_2))\mathbf{q}(t_k). \tag{7.94}$$

Variable $\mathbf{s}_p(t_k)$ needs to appear also in the second equation of (7.85). To this end, we give the following Lemma.

Lemma 15. *Given $\mathcal{U}$ as defined in Appendix A.2, the following holds*

$$\mathbf{s}(t) = \mathcal{U}'\mathbf{s}_p(t) + (\mathbb{I}_{2n} - \mathcal{U}'\mathcal{U})\mathbf{s}(t). \tag{7.95}$$

Proof. Equation $\mathbf{s}(t) = \mathcal{U}'\mathbf{s}_p(t) + (\mathbb{I}_{2n} - \mathcal{U}'\mathcal{U})\mathbf{s}(t)$ is equivalent to

$$\mathcal{U}'\mathcal{U}\mathbf{s}(t) = \mathcal{U}'\mathbf{s}_p(t),$$

which, by (7.88), always holds. □

In the second matricial equation of (7.85), we rewrite $\mathbf{s}(t)$ as in (7.95). This implies

$$\mathbf{q}(t_{k+1}) = \Big((\mathbb{I}_n - D_n^\sigma(k)) \otimes \mathbb{I}_2\Big)\Big(\mathcal{U}'\mathbf{s}_p(t_k) + (\mathbb{I}_{2n} - \mathcal{U}'\mathcal{U})\mathbf{s}(t_k)\Big) + \Big((\mathbb{I}_n - D_n^\sigma(k)) \otimes \mathbb{I}_2\Big)\mathbf{q}(t_k). \tag{7.96}$$

By focusing on the first addendum on the right-hand-side of the equation, by Corollary A.2.1 in Appendix A.2,[5]

$$\begin{aligned}&\Big((\mathbb{I}_n - D_n^\sigma(k)) \otimes \mathbb{I}_2\Big)\Big(\mathcal{U}'\mathbf{s}_p(t_k) + (\mathbb{I}_{2n} - \mathcal{U}'\mathcal{U})\mathbf{s}(t_k)\Big) = \\ &= \Big((\mathbb{I}_n - D_n^\sigma(k)) \otimes \mathbb{I}_2\Big)\Big(\mathcal{U}'\mathbf{s}_p(t_k) + (\frac{1}{n}(\mathbf{1}_n \otimes \mathbb{I}_2)(\mathbf{1}_n \otimes \mathbb{I}_2)')\mathbf{s}(t_k)\Big) = \\ &= \Big(\Big((\mathbb{I}_n - D_n^\sigma(k))U'\Big) \otimes \mathbb{I}_2\Big)\,\mathbf{s}_p(t_k).\end{aligned} \tag{7.97}$$

[5] Consider also that $\Big(\Big((\mathbb{I}_n - D_n^\sigma(k))\Big) \otimes \mathbb{I}_2\Big)\,(\mathbf{1}_n \otimes \mathbb{I}_2)(\mathbf{1}_n \otimes \mathbb{I}_2)' = \mathbf{0}_{2n\times 2n}$, since $D_n^\sigma(k)\mathbf{1}_n = \mathbf{1}_n$.

Finally, by the latter,

$$\mathbf{q}(t_{k+1}) = \Big(\Big((\mathbb{I}_n - D_n^\sigma(k))U'\Big) \otimes \mathbb{I}_2\Big)\,\mathbf{s}_p(t_k) + \Big((\mathbb{I}_n - D_n^\sigma(k)) \otimes \mathbb{I}_2\Big)\mathbf{q}(t_k). \tag{7.98}$$

By bringing (7.94) and (7.98) together into compact form, one obtains

$$\begin{bmatrix}\mathbf{s}_p(t_k^+)\\ \mathbf{q}(t_k^+)\end{bmatrix} = \begin{bmatrix}Q^\sigma(k) \otimes \mathbb{I}_2 & (Q^\sigma(k)U) \otimes \mathbb{I}_2\\ \Big((\mathbb{I}_n - D_n^\sigma(k))U'\Big) \otimes \mathbb{I}_2 & (\mathbb{I}_n - D_n^\sigma(k)) \otimes \mathbb{I}_2\end{bmatrix}\begin{bmatrix}\mathbf{s}_p(t_k)\\ \mathbf{q}(t_k)\end{bmatrix}. \tag{7.99}$$

This latter describes, in terms of $[\mathbf{s}_p(t)', \mathbf{q}(t)']'$, the jump dynamics (7.77). We bring also the flow dynamics (7.82) to a compact form in terms of $[\mathbf{s}_p(t)', \mathbf{q}(t)']'$. The condition

$$\forall t \in (t_k, t_{k+1}],\ k \in \mathbb{N}_0,\ \forall i \in \mathcal{N}, \qquad \dot{\mathbf{s}}_i(t) = \mathbf{0}_2$$

implies

$$\forall t \in (t_k, t_{k+1}],\ k \in \mathbb{N}_0, \qquad \dot{\mathbf{s}}_p(t) = \mathbf{0}_{2(n-1)}, \tag{7.100}$$

which simply implies

$$\forall k \in \mathbb{N}_0, \qquad \mathbf{s}_p(t_{k+1}) = \mathbf{s}_p(t_k^+). \tag{7.101}$$

In order to bring into compact form the first equation of (7.82), consider what shown on pg. 134. By considering (7.75) in terms of $\mathbf{q}_i(t)$ for the jump dynamics (7.82), one obtains

$$\forall t \in (t_k, t_{k+1}],\ \forall i \in \mathcal{N},\ \exists \varepsilon_i \in \mathbb{R}_{>0},\ \exists \rho_i \in \mathbb{R}_{>0} : \\ \|\mathbf{q}_i(t)\| \leq \varepsilon_i \|\mathbf{q}_i(t_k)\| e^{-\rho_i (t - t_k)}. \tag{7.102}$$

In matrix form, the latter becomes, $\forall k \in \mathbb{N}_0$,

$$\begin{bmatrix} \|\mathbf{q}_1(t_{k+1})\| \\ \dots \\ \|\mathbf{q}_n(t_{k+1})\| \end{bmatrix} \leq \\ \begin{bmatrix} \varepsilon_1 e^{-\rho_1(\Delta_k)} & & 0 \\ & \ddots & \\ 0 & & \varepsilon_n e^{-\rho_n(\Delta_k)} \end{bmatrix} \begin{bmatrix} \|\mathbf{q}_1(t_k^+)\| \\ \dots \\ \|\mathbf{q}_n(t_k^+)\| \end{bmatrix}. \tag{7.103}$$

By Proposition A.3.1 in Appendix A.3.1, the latter can be rewritten in compact form as,

$$\forall k \in \mathbb{N}_0, \qquad \|\mathbf{q}(t_{k+1})\| = \left\| \begin{bmatrix} \|\mathbf{q}_1(t_{k+1})\| \\ \dots \\ \|\mathbf{q}_n(t_{k+1})\| \end{bmatrix} \right\| \leq \\ \leq \varepsilon e^{-\rho(\Delta_k)} \left\| \begin{bmatrix} \|\mathbf{q}_1(t_k^+)\| \\ \dots \\ \|\mathbf{q}_n(t_k^+)\| \end{bmatrix} \right\| = \varepsilon e^{-\rho(\Delta_k)} \|\mathbf{q}(t_k^+)]\|, \tag{7.104}$$

where $\varepsilon e^{-\rho(\Delta_k)}$ is the maximum diagonal entry of the matrix in (7.103), namely the spectral norm of a diagonal matrix (corresponding to its norm). By merging together (7.101) and (7.104), we obtain the equivalent of the flow dynamics (7.78) expressed in terms of $[\|\mathbf{s}_p(t)\|, \|\mathbf{q}(t)\|]'$.

Lemma 16. *The system achieves formation in the sense of* (7.5) *if (and only if)*

$$\lim_{k \to \infty} \begin{bmatrix} \|\mathbf{s}_p(t_k)\| \\ \|\mathbf{q}(t_k)\| \end{bmatrix} = \mathbf{0}_2. \tag{7.105}$$

Proof. The proof is trivial. □

As shown in the previous chapters, $\forall k \in \mathbb{N}_0$, $D_n^\sigma(k)$ is, by construction, in general non symmetric. By this, also $Q^\sigma(k)$ will be non symmetric in general. Therefore for non-symmetric (non Hermitian) matrices, it can happen that

$$\|Q^\sigma(k)\| > \rho(Q^\sigma(k)). \tag{7.106}$$

This means that $\|Q^\sigma(k)\|$ can also be larger than 1. To prove the convergence to a formation, we need to make use of the concept of *joint spectral radius*, already analyzed in Definition 15 (pg. 76). As in pg. 76, we assume channel coefficients to be quantized. This way, the set $\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}$ was shown to be finite.

Proposition 41. *For an underlying network topology being a sequence of strongly connected digraphs and notwithstanding the presence of an unknown channel, if $\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}$ is a finite set,*

$$\rho(\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}) < 1. \tag{7.107}$$

Proof. By definition of joint spectral radius, see Definition 15,

$$\rho(\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}) := \lim_{k\to\infty} \max_{\varrho} \|Q^\sigma(\varrho(1)) \cdot \cdots \cdot Q^\sigma(\varrho(k))\|, \\ \varrho : \{1,\dots,k\} \to \{1,\dots,R\}, \tag{7.108}$$

with R being the cardinality of the finite set $\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}$. Note that, $\forall k, h \in \mathbb{N}_0$,

$$\|Q^\sigma(k+h) \cdot \cdots \cdot Q^\sigma(k)\| = \\ \|UD_n^\sigma(k+h)U'UD_n^\sigma(k+h-1)U' \cdot \cdots \cdot UD_n^\sigma(k)U'\|. \tag{7.109}$$

By Theorem A.2.1, $\forall k \in \mathbb{N}_0$,

$$UD_n^\sigma(k+1)U'UD_n^\sigma(k)U' = \\ = UD_n^\sigma(k+1)(\mathbb{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n')D_n^\sigma(k)U' = \\ = UD_n^\sigma(k+1)D_n^\sigma(k)U'. \tag{7.110}$$

By inserting the latter into (7.109), one obtains, $\forall k \in \mathbb{N}_0$,

$$\left\| \prod_{t=0}^{h} Q^\sigma(k+h-t) \right\| = \left\| U \prod_{t=0}^{h} D_n^\sigma(k+h-t)U' \right\|. \tag{7.111}$$

By Theorem 4 on page 12, for $D_n^\sigma(k)$ being associated to a strongly connected topology, $\forall k \in \mathbb{N}_0$,

$$\lim_{h\to\infty} \prod_{\{i=1\dots h | D_i \in \{D_n^\sigma(k)\}_{k\in\mathbb{N}_0}\}} D_i = \mathbf{1}_n(\boldsymbol{w}_n^*)',$$

for some $\boldsymbol{w}_n^* \in \mathbb{R}^n$. By this latter, by (7.111), and by (7.108), since $U\mathbf{1}_n = \mathbf{0}_{n-1}$, one obtains that

$$\lim_{k\to\infty} \rho(\{Q^\sigma(k)\}_{k\in\mathbb{N}_0})^k = 0.$$

The latter yields $\rho(\{Q^\sigma(k)\}_{k\in\mathbb{N}_0}) < \rho < 1$,[6] thus concluding the proof. □

[6] In what follows, consider ρ to be a real-number strictly less than 1.

Corollary 13. *For an underlying network topology being a sequence of strongly connected digraphs and notwithstanding the presence of an unknown channel, $\forall k, h \in \mathbb{N}$, for any $1 > \rho > \rho(\{Q^\sigma(k)\}_{k\in\mathbb{N}_0})$ there exists $C \in \mathbb{R}_{>0}$, for which*

$$\|Q^\sigma(k+h)\cdot\cdots\cdot Q^\sigma(k)\| \leq C\rho^h. \tag{7.112}$$

Proof. In (5.32), on page 78 we have shown that, $\forall x$,

$$\|Q^\sigma(k+h)\cdot\cdots\cdot Q^\sigma(k)x\| \leq Cq^h \|x\|.$$

This latter together with the definition of matrix norm yield the proof. □

By merging together (7.99) and (7.101), one obtains, $\forall k \in \mathbb{N}_0$,

$$\mathbf{s}_p(t_{k+1}) = (Q^\sigma(k)\otimes\mathbb{I}_2)\mathbf{s}_p(t_k) + (Q^\sigma(k)U\otimes\mathbb{I}_2)\mathbf{q}(t_k). \tag{7.113}$$

This equation describes the dynamics of $\mathbf{s}_p(t)$ from t_k to t_{k+1}, $\forall k \in \mathbb{N}_0$. On the other hand, the dynamics of $\mathbf{q}(t)$ from t_k to t_{k+1} cannot be expressed in a linear matrix form, but needs to be described with a norm inequality (due to the presence of the nonlinear controller). In fact, by inserting the second line of (7.99) into (7.104), one obtains, $\forall k \in \mathbb{N}_0$,

$$\begin{aligned}\left\|\mathbf{q}(t_{k+1})\right\| \leq \varepsilon e^{-\rho(\Delta_k)} \|\mathbb{I}_n - D_n^\sigma(k)\| \|\mathbf{s}_p(t_k)\| + \\ \varepsilon e^{-\rho(\Delta_k)} \|\mathbb{I}_n - D_n^\sigma(k)\| \|\mathbf{q}(t_k)\|.\end{aligned} \tag{7.114}$$

Equations (7.113)-(7.114) describe the dynamics of the system. By Lemma 16, if both $\left\|\mathbf{q}(t_{k+1})\right\|$ and $\left\|\mathbf{s}_p(t_{k+1})\right\|$ tend to 0, then the system achieves the desired formation.

Lemma 17. *Consider a set of agents with dynamics (7.77)-(7.78) and communication system (7.76). If $\mathfrak{G}$ is a sequence of strongly connected topologies, it always exists a lower-bound to Δ_k, i.e., $\underline{\Delta} \leq \Delta_k$, $k \in \mathbb{N}_0$,[7] such that*

[7] See (7.3).

$$\lim_{k\to\infty} \begin{bmatrix} \|\mathbf{q}(t_k)\| \\ \|\mathbf{s}_p(t_k)\| \end{bmatrix} = \mathbf{0}_2. \tag{7.115}$$

Proof. By (7.3) and (7.114), one gets

$$\left\|\mathbf{q}(t_{k+1})\right\| \leq \gamma \|\mathbf{s}_p(t_k)\| + \gamma \|\mathbf{q}(t_k)\|, \tag{7.116}$$

with

$$\gamma := \varepsilon e^{-\rho(\underline{\Delta})} \max_{k\in\mathbb{N}_0} \left(\|\mathbb{I}_n - D_n^\sigma(k)\|\right). \tag{7.117}$$

By recursively expanding (7.116) one obtains, $\forall k \in \mathbb{N}_0$,

$$\|\mathbf{q}(t_k)\| \leq \gamma^k \|\mathbf{q}(t_0)\| + \sum_{h=1}^{k} \gamma^h \left\|\mathbf{s}_p(t_{k-h})\right\|. \tag{7.118}$$

By operating the same expansion on (7.113), one obtains

$$\begin{aligned}\mathbf{s}_p(t_k) = (Q^\sigma(k-1)\cdot\cdots\cdot Q^\sigma(0)\otimes\mathbb{I}_2)\,\mathbf{s}_p(t_0) + \\ \sum_{h=1}^{k}\left[\left(\prod_{p=1}^{h} Q^\sigma(k-p)\right)\otimes\mathbb{I}_2\right]\mathcal{U}\mathbf{q}(t_{k-h}).\end{aligned} \tag{7.119}$$

Taking the norm of the latter,

$$\|\mathbf{s}_p(t_k)\| \leq \|Q^\sigma(k-1)\cdot\cdots\cdot Q^\sigma(0)\mathbf{s}_p(t_0)\| + \sum_{h=1}^{k}\left\|\left[\left(\prod_{p=1}^{h}Q^\sigma(k-p)\right)\otimes\mathbb{I}_2\right]\mathcal{U}\mathbf{q}(t_{k-h})\right\|,$$

which, by Corollary 13, yields,[8] $\forall k\in\mathbb{N}_0$,

$$\|\mathbf{s}_p(t_k)\| \leq C\rho^k\|\mathbf{s}_p(t_0)\| + C\sum_{h=1}^{k}\rho^h\left\|\mathbf{q}(t_{k-h})\right\|, \tag{7.120}$$

[8] Recall that $\|A\otimes B\| = \|A\|\,\|B\|$, see [47].

for any $\rho > \rho(\{Q^\sigma(k)\}_{k\in\mathbb{N}_0})$. Equations (7.118)-(7.120) can be analyzed to assess the convergence of the system to a formation. Let, $\forall k\in\mathbb{N}_0$, $S_k\in\mathbb{R}_{\geq 0}$ and $Q_k\in\mathbb{R}_{\geq 0}$ be two variables defined as

$$S_k := C\rho^k\|\mathbf{s}_p(t_0)\| + C\sum_{h=1}^{k}\rho^h\left\|\mathbf{q}(t_{k-h})\right\|, \tag{7.121}$$

$$Q_k := \gamma^k\|\mathbf{q}(t_0)\| + \sum_{h=1}^{k}\gamma^h\left\|\mathbf{s}_p(t_{k-h})\right\|, \tag{7.122}$$

with initial conditions $S_0 = \mathbf{s}_p(t_0)$ and $Q_0 = \mathbf{q}(t_0)$. It is clear that, $\forall k\in\mathbb{N}_0$,

$$\|\mathbf{s}_p(t_k)\| \leq S_k, \tag{7.123}$$

and

$$\|\mathbf{q}(t_k)\| \leq Q_k. \tag{7.124}$$

Variables S_k and Q_k at iteration $k+1$ verify that, respectively,

$$\begin{aligned} S_{k+1} &= C\rho^{k+1}\|\mathbf{s}_p(t_0)\| + C\sum_{h=1}^{k+1}\rho^h\left\|\mathbf{q}(t_{k+1-h})\right\| \\ &= \rho\left(C\rho^k\|\mathbf{s}_p(t_0)\| + C\sum_{h=1}^{k}\rho^h\left\|\mathbf{q}(t_{k-h})\right\|\right) + C\rho\|\mathbf{q}(t_k)\| \\ &= \rho S_k + C\rho\|\mathbf{q}(t_k)\| \\ &\leq \rho S_k + C\rho Q_k, \end{aligned} \tag{7.125}$$

and

$$\begin{aligned} Q_{k+1} &= \gamma^{k+1}\|\mathbf{q}(t_0)\| + \sum_{h=1}^{k+1}\gamma^h\left\|\mathbf{s}_p(t_{k+1-h})\right\| \\ &= \gamma\left(\gamma^k\|\mathbf{q}(t_0)\| + \sum_{h=1}^{k}\gamma^h\left\|\mathbf{s}_p(t_{k-h})\right\|\right) + \gamma\|\mathbf{s}_p(t_k)\| \\ &= \gamma Q_k + \gamma\|\mathbf{s}_p(t_k)\| \\ &\leq \gamma Q_k + \gamma S_k. \end{aligned} \tag{7.126}$$

Consider system, $\forall k\in\mathbb{N}_0$,

$$\begin{bmatrix} S_{k+1} \\ Q_{k+1}\end{bmatrix} \leq \mathbf{X}\begin{bmatrix} S_k \\ Q_k\end{bmatrix} \tag{7.127}$$

(element-wise inequality) where

$$\mathbf{X} := \begin{bmatrix} \rho & C\rho \\ \gamma & \gamma \end{bmatrix}. \tag{7.128}$$

$\mathbf{X}$'s eigenvalues, say λ_1 and λ_2 (assume $|\lambda_1| \leq |\lambda_2|$), determine system (7.127)'s convergence. They can be obtained by solving the following equation of variable λ,

$$\lambda^2 - \lambda \cdot \mathrm{tr}(\mathbf{X}) + \det(\mathbf{X}) = 0,$$

which yields

$$\lambda_{1,2} = \frac{\mathrm{tr}(\mathbf{X}) \pm \sqrt{\mathrm{tr}^2(\mathbf{X}) - 4\det(\mathbf{X})}}{2}.$$

Note that both eigenvalues are real since

$$\mathrm{tr}^2(\mathbf{X}) - 4\det(\mathbf{X}) = \rho^2 + \gamma^2 + 2\rho\gamma(2C-1),$$

and being $C > 0$,

$$\mathrm{tr}^2(\mathbf{X}) - 4\det(\mathbf{X}) = \rho^2 + \gamma^2 + 2\rho\gamma(2C-1) > (\rho - \gamma)^2 \geq 0.$$

Asymptotic stability is then guaranteed for $-1 < \lambda_1 \leq \lambda_2 < 1$. It is also clear from above (since $\mathrm{tr}(\mathbf{X}) > 0$) that $\lambda_2 > 0$ and $\lambda_2 > |\lambda_1|$. Thus, asymptotic stability is guaranteed for $\lambda_2 < 1$, namely,

$$\sqrt{\mathrm{tr}^2(\mathbf{X}) - 4\det(\mathbf{X})} < 2 - \mathrm{tr}(\mathbf{X}).$$

If $\mathrm{tr}(\mathbf{X}) = \rho + \gamma < 2$,[9] the latter inequality yields

[9] This is trivial to verify, as $\rho < 1$ and γ can be controlled by choosing $\underline{\Delta}$.

$$0 < \gamma < \frac{1-\rho}{1+\rho(C-1)}.$$

Being $\rho < 1$ and $C > 0$, the interval

$$\left(0,\ \frac{1-\rho}{1+\rho(C-1)}\right)$$

is always nonempty. This shows that there always exists $\underline{\Delta} \in \mathbb{R}_{>0}$ [10] satisfying

[10] $\forall k \in \mathbb{N}_0,\ \underline{\Delta} \leq \Delta_k$ which implies, $\forall k \in \mathbb{N}_0,\ \varepsilon e^{-\rho(\Delta_k)} \leq \varepsilon e^{-\rho(\underline{\Delta})}$.

$$\varepsilon e^{-\rho(\underline{\Delta})} \max_{k\in\mathbb{N}_0} (\|\mathbb{I}_n - D_n^{\sigma}(k)\|) \in \left(0,\ \frac{1-\rho}{1+\rho(C-1)}\right).$$

By choosing such a $\underline{\Delta}$, we obtain

$$\lim_{k\to\infty} \begin{bmatrix} S_k \\ Q_k \end{bmatrix} = \mathbf{0}_2.$$

In case the latter holds, by (7.123)-(7.124), also (7.115) is verified. This concludes the proof. □

Theorem 14 is finally proven by employing Lemma 16 and Lemma 17. □

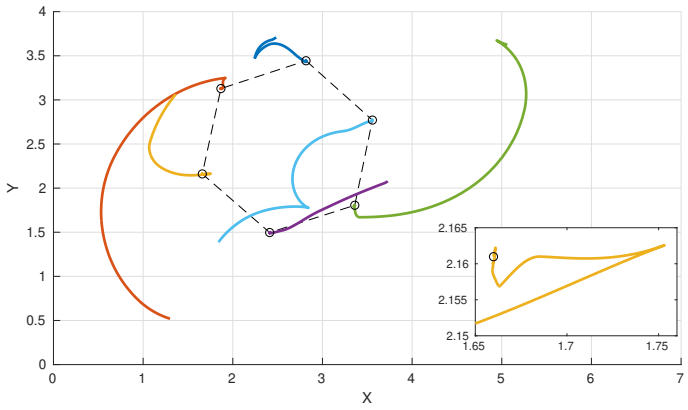


Figure 7.6: Trajectories in the space of agents in $\mathcal{N}$ seeking for a formation.

7.2.2.4 *Simulations*

A system with $n = 6$ agents, with the same initial conditions, underlying network topology, and channel realization as the experiment in Section 7.1.3 is simulated. This time, the control strategy is (7.77)-(7.78). We show the impact of such a nonlinear control approach if combined with the exploitation of interference. Figure 7.6 illustrates the result. We consider a sequence of 20 update steps. One can see

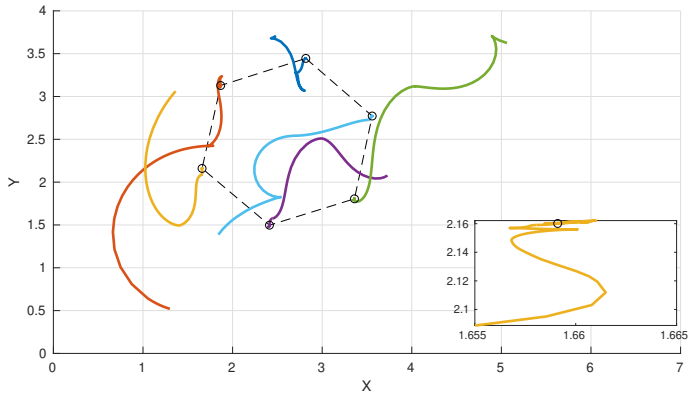


Figure 7.7: Trajectories in the space of agents in $\mathcal{N}$ seeking for a formation. This simulation considers a larger $\underline{\Delta}$.

that, compared to the linearized approach of Section 7.1.3, trajectories here are smoother. Also when approaching the final converging position, robots keep manoeuvring (this is clear from the magnifying box in the plot).

Starting with the same initial conditions and channel realization, Figure 7.7 shows what happens if $\underline{\Delta}$ is increased (of 5 times). Simulation is also run for 20 consecutive update steps. Trajectories are more precise (look, e.g., at the yellow agent and compare it to Figure 7.6). This, in fact, is the case if agents have more time to track the agreed positions after the jump dynamics.

Finally, by picking $\underline{\Delta}$ considerably smaller, Figure 7.8 illustrates what happens. Simulations are run for 20 consecutive update steps. Being the lower bound on Δ_k too small, a formation cannot be achieved.

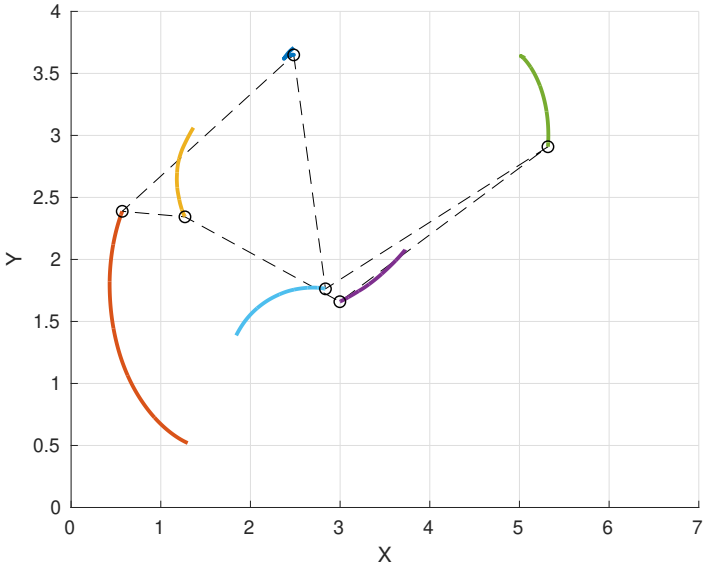


Figure 7.8: Trajectories in the space of agents in $\mathcal{N}$ seeking for a formation. This simulation considers a very small $\underline{\Delta}$, thus not allowing a formation to be achieved.

8
Conclusion

8.1 Contribution

The main contribution of this thesis is the development of a family of consensus protocols that allow a multi-agent system to reach an agreement by exploiting the interference property of the wireless channel. In standard approaches, interference needs to be avoided, in order to provide each agent with the exact knowledge of information states of other agents. However, this requires some communication resources that, traditionally, have not been taken into account for quantifying the energy cost of this or that consensus protocol. Our strategy makes it possible to exploit interference. All agents can simultaneously broadcast to the channel and retrieve a faded and superposed signal that can be used for reaching consensus. Such a signal, in standard practice, was labelled as "corrupted". In fact, the fading effect is unknown a priori and the superposition sums up all such signals, making it impossible to reconstruct each individual one. Our aim is to reconsider and appreciate this corrupted signal, because it carries some information that can be used for reaching an agreement. This technology is, at the current state of the art, explored only theoretically. However, the possible ways to apply it to common practical problems are numerous; two of them, i.e., solving algebraic equations over the channel and achieving a formation in space, have been investigated.

After having provided in Chapter 2 the theoretical tools needed for designing and testing consensus protocols, two novel ways of using consensus for the automation of road traffic have been presented in Chapter 3. In fact, traffic is a very concrete example of a multi-agent system in which agents can communicate and take individual decisions which affect other agents. In the first use case, we propose to use consensus for letting vehicles agree on crossing priorities for a road intersection. This, together with an MPC that avoids collisions, constitutes a novel approach to the automation of road intersections. In the second use case, the scenario is the highway. Vehicles can use average consensus for agreeing on the lane speed. After that, to decide where a vehicle aiming at changing lane can cut in, we propose the usage of a consensus-based auction algorithm. As for the previous case, this algorithm, together with an MPC for collision avoidance, provides a novel approach for automating highways.

These solutions are based on standard communication systems. For presenting our novel interference-base strategies, in Chapter 4 we give a communication-theoretical brief overview of a wireless channel in which transmitters simultaneously access the same frequency at the same time. Such model is called WMAC and is used for developing the solution proposed in Section 5, where an average consensus protocol allows to exploit interference for reaching an agreement. This drastically reduces the wireless resources needed to reach the agreement, especially for large networks. Although the agreement value depends on the realization of the channel (and is not known a priori), it is contained in the convex hull of the set that groups initial information states. Finally, such approaches to average consensus are used for solving linear algebraic equations over the channel. This represents a useful application of the strategy we have proposed, since it allows for solving an equation, while maintaining privacy, i.e., agents can no longer access the individual information states of other agents (and their historical values).

Chapter 6 considers a nonlinear consensus protocol, i.e, max-consensus. For exploiting the interference property of the channel and guaranteeing finite-time agreement, agents have to employ a switching consensus protocol that is presented and proven to converge. Comparisons with standard approaches show that a numerous amount of wireless resources can be saved by exploiting interference, especially in large networks.

The most relevant application of consensus is controlling agents to a formation. In Chapter 7, we have proposed a way to exploit interference for achieving a formation in space. In the first scenario, we consider robots with single-integrator dynamics, in which movements on the two axes are independent. After that, a realistic nonholonomic (bicycle) model for robot dynamics is considered. Controlling a group of robots with that dynamics can be done via two diverse approaches. The first presented approach is by feedback linearization, the second one by employing a nonlinear controller. In both case, we have proven that agents can achieve the formation by exploiting the interference property of the channel.

8.2 Future work

Concerning theoretical advancements, future work might consider different kind of consensus protocols, less notorious in the consensus community, e.g., median consensus or consensus for Kuramoto oscillators. The impact of asynchronism can be also of interest. However, to understand the extent of asynchronism in this application, we need to have a better practical characterization of the technology.

In fact, the technology and the protocols presented in this thesis have been investigated only under a theoretical point of view. The major consequence of this thesis is to provide a solid theoretical framework for the exploitation of interference for consensus, so that practical experiments can be finally conducted. This is the most ambitious

aspect of future work. Once the algorithms designed in this thesis are practically implemented, the number of possible fields where these can be applied is enormous.

A
Appendix

A.1 Jordan normal form and generalized eigenvectors

Let $A \in \mathbb{R}^{n\times n}$ be a real-valued square matrix. A *generalized eigenvector* for A associated to eigenvalue λ is a vector $\boldsymbol{v} \in \mathbb{C}^n$ for which

$$(A - \lambda \mathbb{I}_n)^k \boldsymbol{v} = 0, \tag{A.1}$$

for some positive integer k. The smallest k for which (A.1) holds is the *generalized eigenvector rank*.

Observation A.1.1. *An eigenvector is a generalized eigenvector of rank* 1.

Definition A.1.1. *Given an eigenvalue λ, vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_r \in \mathbb{R}^n$ form a Jordan chain of length r if $\boldsymbol{v}_1 \neq \mathbf{0}_n$ and*

$$\begin{aligned}
\boldsymbol{v}_{r-1} &= (A - \lambda \mathbb{I}_n)\boldsymbol{v}_r, \\
\boldsymbol{v}_{r-2} &= (A - \lambda \mathbb{I}_n)\boldsymbol{v}_{r-1}, \\
&\cdots \\
\boldsymbol{v}_1 &= (A - \lambda \mathbb{I}_n)\boldsymbol{v}_2, \\
\mathbf{0}_n &= (A - \lambda \mathbb{I}_n)\boldsymbol{v}_1.
\end{aligned} \tag{A.2}$$

Vector $\boldsymbol{v}_1$ is clearly the eigenvector of A associated with λ.

Theorem A.1.1 ([48, pg. 230])**.** *The Jordan chain of A is composed of linearly independent vectors.*

Note that, by Theorem A.1.1, if $\boldsymbol{v}_r \in \mathbb{R}^n$ is a generalized eigenvector of A of rank r, then there are vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{r-1} \in \mathbb{R}^n$ for which

$$\begin{aligned}
A\boldsymbol{v}_1 &= \lambda \boldsymbol{v}_1, \\
A\boldsymbol{v}_2 &= \lambda \boldsymbol{v}_2 + \boldsymbol{v}_1, \\
&\cdots \\
A\boldsymbol{v}_r &= \lambda \boldsymbol{v}_r + \boldsymbol{v}_{r-1}.
\end{aligned} \tag{A.3}$$

In case A fails to be diagonalizable[1], generalized eigenvectors comes into play to "enlarge" the set of linearly independent eigenvectors of A.

[1] When at least one eigenvalue of A has algebraic multiplicity strictly larger than its geometric multiplicity.

Theorem A.1.2. *With regards to Theorem A.1.1, the subspace*

$$\text{span}(\boldsymbol{v}_1, \dots, \boldsymbol{v}_r) \subseteq \mathbb{R}^n$$

has dimension $r \in \mathbb{N}$ and is called Jordan subspace for A.

Definition A.1.2. *Given any (complex) square matrix A, its* Jordan matrix *is J, defined through diagonal blocks, i.e.,*

$$J = \text{diag}\left(J_1, \dots, J_\ell\right),$$

such that, $\forall i = 1, \dots, \ell$, $J_i \in \mathbb{R}^{m(i)\times m(i)}$, with $m(i)$ algebraic multiplicity of λ_i, and

$$J_i = \begin{bmatrix} \lambda_i & 1 & 0 & \dots & 0 \\ 0 & \lambda_i & 1 & \dots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 1 \\ 0 & \dots & \dots & 0 & \lambda_i \end{bmatrix}.$$

Theorem A.1.3 ([48, pg. 237])**.** *Any (complex) square matrix A is similar to its* Jordan matrix*, i.e.,*

$$A = PJP^{-1}, \tag{A.4}$$

where P's columns are a basis of generalized eigenvectors of A ordered according to the corresponding eigenvalue on J's diagonal and arranged in chains.

For a diagonalizable matrix $A \in \mathbb{R}^{n\times n}$, we have, $\forall p \in \mathbb{N}$,

$$A^p \boldsymbol{v}_n = \lambda^p \boldsymbol{v}_n, \tag{A.5}$$

with $(\lambda, \boldsymbol{v}_n)$ eigenpair of B. What happens when $(\lambda, \boldsymbol{v}_n)$ is a generalized eigenpair of A is the topic of the following theorem.

Theorem A.1.4. *A real-valued square matrix A of dimension n is given, together with one generalized eigenpair, i.e., $(\lambda, \boldsymbol{v}_p)$ (p is the rank of the generalized eigenvector). We have, $\forall \alpha \in \mathbb{N}$,*

$$A^\alpha \boldsymbol{v}_p = \sum_{\ell=0}^{\min(\alpha, p-1)} \binom{\alpha}{\ell} \lambda^{\alpha-\ell} \boldsymbol{v}_{p-\ell}. \tag{A.6}$$

Proof. Let $B = (A - \lambda \mathbb{I}_n)$. By Definition A.1.1,

$$B\boldsymbol{v}_p = \boldsymbol{v}_{p-1} \tag{A.7}$$

and

$$B^p \boldsymbol{v}_p = \boldsymbol{0}_n. \tag{A.8}$$

By the Newton's generalized binomial theorem,

$$A^\alpha \boldsymbol{v}_p = (\lambda \mathbb{I}_n + B)^\alpha \boldsymbol{v}_p = \sum_{\ell=0}^{\alpha} \binom{\alpha}{\ell} (\lambda \mathbb{I}_n)^{\alpha-\ell} B^j \boldsymbol{v}_p, \tag{A.9}$$

since $\lambda \mathbb{I}_n$ and B commute,

$$\sum_{\ell=0}^{\alpha} \binom{\alpha}{\ell} (\lambda \mathbb{I}_n)^{\alpha-\ell} B^\ell \boldsymbol{v}_p = \sum_{\ell=0}^{\min(\alpha, p-1)} \binom{\alpha}{\ell} \lambda^{\alpha-\ell} B^\ell \boldsymbol{v}_p, \tag{A.10}$$

and, by (A.7)-(A.8),

$$\sum_{\ell=0}^{\min(\alpha,p-1)} \binom{\alpha}{\ell} \lambda^{\alpha-\ell} B^{\ell} \boldsymbol{v}_p = \sum_{\ell=0}^{\min(\alpha,p-1)} \binom{\alpha}{\ell} \lambda^{\alpha-\ell} \boldsymbol{v}_p[p-\ell]. \quad \text{(A.11)}$$

This concludes the proof. □

A.2 Nullifying Matrix

Let $U \in \mathbb{R}^{(n-1)\times n}$ be a full-rank projection matrix from $\mathbb{R}^n$ onto $\mathbb{R}^{n-1}$, such that the n-dimensional vector of ones spans the kernel of U and the columns of U form an orthonormal set, i.e., formally,

$$U\mathbf{1}_n = \mathbf{0}_{n-1} \quad \text{(A.12)}$$

and

$$UU' = \mathbb{I}_{n-1.} \quad \text{(A.13)}$$

Additionally, given $p \in \mathbb{N}$, let's define $\mathcal{U}$ as

$$\mathcal{U} := U \otimes \mathbb{I}_p, \quad \text{(A.14)}$$

where $\otimes$ is the Kronecker product. In the following, we analyze some properties of matrix U.

Theorem A.2.1. *Given the full-rank matrix $U \in \mathbb{R}^{(n-1)\times n}$ such that (A.12)-(A.13) hold,*

$$(\mathbb{I}_n - U'U) = \frac{1}{n}\mathbf{1}_n\mathbf{1}_n'. \quad \text{(A.15)}$$

Proof. Choose $\alpha = \frac{1}{\sqrt{n}}$. One has

$$\begin{bmatrix} U \\ \alpha\mathbf{1}_n' \end{bmatrix} [U', \alpha\mathbf{1}_n] = \begin{bmatrix} UU' & \alpha U\mathbf{1}_n \\ \alpha\mathbf{1}_n'U' & \alpha^2\mathbf{1}_n'\mathbf{1}_n \end{bmatrix} = \mathbb{I}_n, \quad \text{(A.16)}$$

because of (A.12),(A.13) and $\mathbf{1}_n'\mathbf{1}_n = n$. Hence $\begin{bmatrix} U \\ \alpha\mathbf{1}_n' \end{bmatrix}$ is the inverse of $[U', \alpha\mathbf{1}_n]$ and

$$[U', \alpha\mathbf{1}_n] \begin{bmatrix} U \\ \alpha\mathbf{1}_n' \end{bmatrix} = U'U + \alpha^2\mathbf{1}_n\mathbf{1}_n' = \mathbb{I}_n. \quad \text{(A.17)}$$

Therefore:

$$\mathbb{I}_n - U'U = \alpha^2\mathbf{1}_n\mathbf{1}_n' \quad \text{(A.18)}$$

$$= \frac{1}{n}\mathbf{1}_n\mathbf{1}_n'. \quad \text{(A.19)}$$

□

Observation A.2.1. *Matrix $(\mathbb{I}_n - U'U)$ is row-stochastic.*

Lemma A.2.1. *Given $\mathcal{U}$ defined in (A.14),*

$$\mathcal{U}\mathbf{1}_{pn} = \mathbf{0}_{p(n-1)} \quad \text{(A.20)}$$

and

$$\mathcal{U}(\mathbf{1}_n \otimes \mathbb{I}_p) = \mathbf{0}_{(n-1)p\times p}\,. \quad \text{(A.21)}$$

Proof.

$$\mathbf{1}_{pn} = \mathbf{1}_n \otimes \mathbf{1}_p.$$

Hence

$$\begin{aligned}\mathcal{U}\mathbf{1}_{pn} &= (U \otimes \mathbb{I}_p)(\mathbf{1}_n \otimes \mathbf{1}_p) \\ &= (U\mathbf{1}_n) \otimes (\mathbb{I}_p\mathbf{1}_p) \\ &= \mathbf{0}_{(n-1)p}\,.\end{aligned}$$

Similarly,

$$\begin{aligned}\mathcal{U}(\mathbf{1}_n \otimes \mathbb{I}_p) &= (U \otimes \mathbb{I}_p)(\mathbf{1}_n \otimes \mathbb{I}_p) \\ &= U\mathbf{1}_n \otimes \mathbb{I}_p \\ &= \mathbf{0}_{n-1} \otimes \mathbb{I}_p \\ &= \mathbf{0}_{(n-1)p\times p}\,.\end{aligned}$$

□

Corollary A.2.1. *Given $\mathcal{U}$ defined in (A.14),*

$$\mathcal{U}\mathcal{U}' = \mathbb{I}_{(n-1)p},$$

and

$$(\mathbb{I}_{pn} - \mathcal{U}'\mathcal{U}) = \frac{1}{n}(\mathbf{1}_n \otimes \mathbb{I}_p)(\mathbf{1}_n \otimes \mathbb{I}_p)'\,.$$

Proof. Using the properties of the Kronecker product and (A.13),

$$\begin{aligned}\mathcal{U}\mathcal{U}' &= (U \otimes \mathbb{I}_p)(\mathcal{U}' \otimes \mathbb{I}_p) \\ &= (\mathcal{U}\mathcal{U}') \otimes \mathbb{I}_p \\ &= \mathbb{I}_{n-1} \otimes \mathbb{I}_p \\ &= \mathbb{I}_{(n-1)p}\,. \qquad \text{(A.22)}\end{aligned}$$

Furthermore, with $\alpha = \frac{1}{\sqrt{n}}$, because of A.22 and Lemma A.2.1,

$$\begin{aligned}\begin{bmatrix}\mathcal{U} \\ \alpha(\mathbf{1}_n \otimes \mathbb{I}_p)'\end{bmatrix}[\mathcal{U}',\ \alpha(\boldsymbol{n}_n \otimes \mathbb{I}_p)] &= \begin{bmatrix}\mathcal{U}\mathcal{U}' & \alpha\mathcal{U}(\mathbf{1}_n \otimes \mathbb{I}_p) \\ \alpha(\mathcal{U}(\mathbf{1}_n \otimes \mathbb{I}_p))' & (\mathbf{1}_n \otimes \mathbb{I}_p)'(\mathbf{1}_n \otimes \mathbb{I}_p)\end{bmatrix} \\ &= \begin{bmatrix}\mathbb{I}_{(n-1)p} & \mathbf{0}_{(n-1)p\times p} \\ \mathbf{0}_{(n-1)p\times p} & \dfrac{\alpha^2(\mathbf{1}_n'\mathbf{1}_n \otimes \mathbb{I}_p)}{\mathbb{I}_p}\end{bmatrix} \\ &= \mathbb{I}_{np}\,.\end{aligned}$$

Therefore,

$$\begin{aligned}[\mathcal{U}',\ \alpha(\boldsymbol{n}_n \otimes \mathbb{I}_p)]\begin{bmatrix}\mathcal{U} \\ \alpha(\mathbf{1}_n \otimes \mathbb{I}_p)'\end{bmatrix} &= \mathcal{U}'\mathcal{U} + \frac{1}{n}(\mathbf{1}_n \otimes \mathbb{I}_p)(\mathbf{1}_n \otimes \mathbb{I}_p)' \\ &= \mathbb{I}_{np}\,.\end{aligned}$$

This proves the second part of this corollary. □

Proposition A.2.1. *Given $U \in \mathbb{R}^{(n-1)\times n}$ such that (A.13) holds,*

$$\|U'\| = 1, \qquad \text{(A.23)}$$

where the matrix norm is induced by the Euclidean norm.

Proof.

$$\|U'\|^2 = \lambda_{max}(UU') = 1\,. \tag{A.24}$$

□

Corollary A.2.2. *Given matrix $\mathcal{U}$ defined in (A.14),*

$$\|\mathcal{U}\| = 1, \tag{A.25}$$

where the matrix norm is induced by the Euclidean norm.

Proof.

$$\|\mathcal{U}\|^2 = \lambda_{max}(\mathcal{U}'\mathcal{U}) = \lambda_{max}(\mathbb{I}_{np} - \frac{1}{n}(\mathbf{1}_n \otimes \mathbb{I}_p)(\mathbf{1}_n \otimes \mathbb{I}_p)') = 1\,.$$

□

Proposition A.2.2. *Let D_n^σ be a primitive row-stochastice matrix. Let $U \in \mathbb{R}^{(n-1)\times n}$ such that (A.12)-(A.13) holds. Then,*

$$\exists!\, Q^\sigma \in \mathbb{R}^{(n-1)\times(n-1)} :\; UD_n^\sigma = Q^\sigma U, \tag{A.26}$$

and moreover

$$\text{spectrum}\,(D_n^\sigma) = \{1\} \cup \text{spectrum}\,(Q^\sigma)\,. \tag{A.27}$$

Proof. The proof is given in [40, Eq. (16)-(17)]. □

A.3 Norms

A.3.1 Norm properties

The following results concern vector and matrix norms.

Proposition A.3.1. *A set of p-dimensional vectors, i.e. $\{\boldsymbol{x}_i \in \mathbb{R}^p \mid i = 1 \dots n\}$, is given. Group them into a vector $\boldsymbol{x} := [\boldsymbol{x}_1' \dots \boldsymbol{x}_n']'$. The following is always true:*

$$\|\boldsymbol{x}\| = \left\| \begin{bmatrix} \|\boldsymbol{x}_1\| \\ \dots \\ \|\boldsymbol{x}_n\| \end{bmatrix} \right\|, \tag{A.28}$$

where $\|\cdot\|$ is the Euclidean norm.

Proof. By definition of Euclidean norm,

$$\|\boldsymbol{x}_i\| = \sqrt{\sum_{j=1}^{p} [\boldsymbol{x}_i]_j^2}. \tag{A.29}$$

Also,

$$\left\| \begin{bmatrix} \|\boldsymbol{x}_1\| \\ \dots \\ \|\boldsymbol{x}_n\| \end{bmatrix} \right\| = \sqrt{\sum_{i=1}^{n} \|\boldsymbol{x}_i\|^2}. \tag{A.30}$$

By substituting (A.29) into (A.30) one obtains

$$\left\| \begin{bmatrix} \|\boldsymbol{x}_1\| \\ \dots \\ \|\boldsymbol{x}_n\| \end{bmatrix} \right\| = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{p} [\boldsymbol{x}_i]_j^2}, \tag{A.31}$$

which is the definition of $\|\boldsymbol{x}\|$. This concludes the proof. □

A.3.2 *Mixed norms*

A mixed-matrix and a mixed-vector norms are *compatible* if

$$\|A\boldsymbol{v}\|_{2,\infty} \leq |||A|||_{2,\infty} \|\boldsymbol{v}\|_{2,\infty}\,, \tag{A.32}$$

for any matrix $A \in \mathbb{R}^{dm\times dm}$ and any vector $\boldsymbol{v} \in \mathbb{R}^{dm}$. In fact, by the definition of $\|\cdot\|_{2,\infty}$ and by the triangular inequality for $\|\cdot\|_2$,

$$\begin{aligned}
\|A\boldsymbol{v}\|_{2,\infty} &= \max_{i=1\dots m}\left(\left\|\sum_{\ell=1}^{m}\tilde{A}_{i\ell}\tilde{\boldsymbol{v}}_\ell\right\|_2\right)\\
&\leq \max_{i=1\dots m}\left(\sum_{\ell=1}^{m}\left\|\tilde{A}_{i\ell}\right\|_2\|\tilde{\boldsymbol{v}}_\ell\|_2\right)\\
&\leq \max_{i=1\dots m}\left(\sum_{\ell=1}^{m}\left\|\tilde{A}_{i\ell}\right\|_2\right)\max_{\ell=1\dots m}\left(\|\tilde{\boldsymbol{v}}_\ell\|_2\right)\\
&= |||A|||_{2,\infty}\|\boldsymbol{v}\|_{2,\infty}\,.
\end{aligned}$$

This proves that the presented mixed-matrix and mixed-vector norms are compatible. Moreover, we aim at showing that $|||\cdot|||_{2,\infty}$ is *induced* by $\|\cdot\|_{2,\infty}$. By [37, Def. 5.6.1], this is the case if

$$|||A|||_{2,\infty} = \sup_{\|\boldsymbol{v}\|_{2,\infty}\neq 0}\frac{\|A\boldsymbol{v}\|_{2,\infty}}{\|\boldsymbol{v}\|_{2,\infty}}. \tag{A.33}$$

Since (A.32) is true, condition (A.33) is proven if $\exists\boldsymbol{v}$ for which we get

$$|||A|||_{2,\infty} = \frac{\|A\boldsymbol{v}\|_{2,\infty}}{\|\boldsymbol{v}\|_{2,\infty}}. \tag{A.34}$$

To verify this, let's expand the mixed-matrix norm, so that

$$|||A|||_{2,\infty} = \max_{i=1\dots m}\left(\sum_{\ell=1}^{m}\left\|\tilde{A}_{i\ell}\right\|_2\right). \tag{A.35}$$

Since the spectral norm for matrix is induced by the l_2 vector norm,

$$\max_{i=1\dots m}\left(\sum_{\ell=1}^{m}\left\|\tilde{A}_{i\ell}\right\|_2\right) = \max_{i=1\dots m}\left(\sum_{\ell=1}^{m}\left\|\tilde{A}_{i\ell}\tilde{\boldsymbol{w}}_\ell\right\|_2\right), \tag{A.36}$$

for some $\boldsymbol{w}_\ell$, $\ell = 1\dots m$, such that $\|\tilde{\boldsymbol{w}}_\ell\|_2 = 1$, $\forall\ell = 1\dots m$. By the triangular inequality applied to (A.36),

$$\max_{i=1\dots m}\left(\sum_{\ell=1}^{m}\left\|\tilde{A}_{i\ell}\tilde{\boldsymbol{w}}_\ell\right\|_2\right) \geq \max_{i=1\dots m}\left(\left\|\sum_{\ell=1}^{m}\tilde{A}_{i\ell}\tilde{\boldsymbol{w}}_\ell\right\|_2\right), \tag{A.37}$$

where the right-hand side equals $\|A\boldsymbol{w}\|_{2,\infty}$. Also, by definition of l_2/l_∞ vector norm,

$$\|\boldsymbol{w}\|_{2,\infty} = \max_{\ell=1\dots m}\left(\|\tilde{\boldsymbol{w}}_\ell\|_2\right) = 1,$$

since $\|\tilde{\boldsymbol{w}}_\ell\|_2 = 1$, $\forall\ell = 1\dots m$.

By merging all these considerations, one obtains

$$|||A|||_{2,\infty} \geq \frac{\|A\boldsymbol{w}\|_{2,\infty}}{\|\boldsymbol{w}\|_{2,\infty}}. \tag{A.38}$$

By the latter and (A.32), equation (A.34) is verified.

A.4 Exponential Stabilization of Nonholonomic Agents

Consider an agent with dynamics (7.61), equivalently (7.68) in matrix form. The controller designed in [25] is presented in this Appendix. It guarantees exponential stabilization without the need of infinitely fast switching (as, instead, done by sliding controllers).

Define the following variables, $\forall t \in \mathbb{R}_{\geq 0}$:

$$\theta_d(t) := \begin{cases} 2 \arctan \frac{y_i(t)}{x_i(t)} & \text{if } (x_i(t), y_i(t)) \neq (0,0) \\ 0 & \text{otherwise} \end{cases}, \tag{A.39}$$

$$r_i(t) = r(x_i(t), y_i(t)) := \frac{x_i^2(t) + y_i^2(t)}{2y_i(t)}, \tag{A.40}$$

$$a_i(t) = a_i(x_i(t), y_i(t)) := r_i(t)\theta_d(t), \tag{A.41}$$

$$\alpha_i(t) := \theta_i(t) - \theta_d(t), \tag{A.42}$$

$$\beta_i(t) := \frac{y_i(t)}{x_i(t)}. \tag{A.43}$$

These variables[2] will be used for designing the closed-loop controller. Define the following quantities, $\forall t \in \mathbb{R}_{\geq 0}$,[3]

$$b_1(t) := \cos\theta_i \left(\frac{\theta_d}{\beta_i} - 1 \right) + \sin\theta_i \left(\frac{\theta_d}{2}(1 - \frac{1}{\beta_i^2}) + \frac{1}{\beta_i} \right) \tag{A.44}$$

$$b_2(t) := \cos\theta_i \frac{2\beta_i}{(1+\beta_i^2)x_i} - \sin\theta_i \frac{2}{(1+\beta_i^2)x_i}. \tag{A.45}$$

The closed-loop control law designed in [25] is, $\forall t \in \mathbb{R}_{\geq 0}$,

$$v_i(t) = -\gamma b_1(t) a_i(t) \tag{A.46}$$

$$\omega_i(t) = -b_2(t) v_i(t) - k\alpha_i(t), \tag{A.47}$$

with $\gamma > 0$ and $k > 0$.

[2] In [25], $\alpha_i(t)$ is a periodic and piecewise continuous function in $(-\pi, \pi]$.

[3] In the two following definitions, the time dependence is omitted.

Bibliography

[1] 3GPP.org. Enhancement of 3GPP support for V2X scenarios. 3GPP TS 22.186, V16.2.0, 2019.

[2] J. Almeida, C. Silvestre, and A. M. Pascoal. Continuous-time consensus with discrete-time communications. *Systems & Control Letters*, 61(7):788–796, 2012.

[3] B. D. Anderson, S. Mou, A. S. Morse, and U. Helmke. Decentralized gradient algorithm for solution of a linear equation. *Numerical Algebra, Control & Optimization*, 6(3):319–328, 2016.

[4] C. Asensio-Marco and B. Beferull-Lozano. Energy efficient consensus over complex networks. *IEEE Journal of Selected Topics in Signal Processing*, 9(2):292–303, 2015.

[5] K. J. Aström and R. M. Murray. *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.

[6] K. J. Åström and B. Wittenmark. *Computer-controlled systems: theory and design*. Courier Corporation, 2013.

[7] K. M. Audenaert. On a norm compression inequality for 2xn partitioned block matrices. *arXiv preprint math/0702186*, 2007.

[8] G. Baldan and S. Zampieri. An efficient quantization algorithm for solving average-consensus problems. In *2009 European Control Conference (ECC)*, pages 761–766. IEEE, 2009.

[9] G. Bardaro, L. Bascetta, E. Ceravolo, M. Farina, M. Gabellone, and M. Matteucci. Mpc-based control architecture of an autonomous wheelchair for indoor environments. *Control Engineering Practice*, 78:160–174, 2018.

[10] L. D. Baskar, B. De Schutter, J. Hellendoorn, and Z. Papp. Traffic control and intelligent vehicle highway systems: a survey. *IET Intelligent Transport Systems*, 2011.

[11] L. D. Baskar, B. De Schutter, and H. Hellendoorn. Traffic management for automated highway systems using model-based predictive control. *IEEE Transactions on Intelligent Transportation Systems*, 13(2):838–847, 2012.

[12] I. Bjelaković, M. Frey, and S. Stańczak. Distributed approximation of functions over fast fading channels with applications to

distributed learning and the max-consensus problem. In *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1146–1153. IEEE, 2019.

[13] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis. Convergence in multiagent coordination, consensus, and flocking. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 2996–3000. IEEE, 2005.

[14] T. Borzone, I.-C. Morărescu, M. Jungers, M. Boc, and C. Janneteau. Hybrid framework for consensus in fleets of nonholonomic robots. In *2018 Annual American Control Conference (ACC)*, pages 4299–4304. IEEE, 2018.

[15] R. W. Brockett et al. Asymptotic stability and feedback stabilization. *Differential geometric control theory*, 27(1):181–191, 1983.

[16] C. Cachin and M. Vukolić. Blockchain consensus protocols in the wild. *arXiv preprint arXiv:1707.01873*, 2017.

[17] C. H. Caicedo-Nunez and M. Zefran. Distributed task assignment in mobile sensor networks. *IEEE transactions on automatic control*, 56(10):2485–2489, 2011.

[18] G. R. Campos, P. Falcone, H. Wymeersch, R. Hult, and J. Sjöberg. Cooperative receding horizon conflict resolution at traffic intersections. In *53rd IEEE Conference on Decision and Control*, pages 2932–2937, 2014.

[19] G. R. Campos, P. Falcone, R. Hult, H. Wymeersch, and J. Sjöberg. Traffic coordination at road intersections: Autonomous decision-making algorithms using model-based heuristics. *IEEE Intelligent Transportation Systems Magazine*, 9 (1):8–21, 2017.

[20] M. Cao, D. A. Spielman, and A. S. Morse. A lower bound on convergence of a distributed network consensus algorithm. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 2356–2361. IEEE, 2005.

[21] M. Cao, A. S. Morse, and B. D. Anderson. Reaching a consensus in a dynamically changing environment: A graphical approach. *SIAM Journal on Control and Optimization*, 47(2):575–600, 2008.

[22] R. Carli, F. Fagnani, P. Frasca, and S. Zampieri. Efficient quantization for average consensus. *arXiv preprint arXiv:0903.1337*, 2009.

[23] H.-L. Choi, L. Brunet, and J. P. How. Consensus-based decentralized auctions for robust task allocation. *IEEE transactions on robotics*, 25(4):912–926, 2009.

[24] A. De Luca, G. Oriolo, and C. Samson. Feedback control of a nonholonomic car-like robot. In *Robot motion planning and control*, pages 171–253. Springer, 1998.

[25] C. C. De Wit and O. Sordalen. Exponential stabilization of mobile robots with nonholonomic constraints. In [*1991*] *Proceedings of the 30th IEEE Conference on Decision and Control*, pages 692–697. IEEE, 1991.

[26] M. di Bernardo, A. Salvi, S. Santini, and A. S. Valente. Third-order consensus in vehicles platoon with heterogeneous time-varying delays. *IFAC-PapersOnLine*, 48(12):358–363, 2015.

[27] E. J. Dudewicz. Introduction to statistics and probability. 1976.

[28] Y. C. Eldar, P. Kuppinger, and H. Bolcskei. Block-sparse signals: Uncertainty relations and efficient recovery. *IEEE Transactions on Signal Processing*, 58(6):3042–3054, 2010.

[29] J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations. *IFAC Proceedings Volumes*, 35(1): 115–120, 2002.

[30] A. Ferrara and C. Vecchio. Controlling a platoon of vehicles via a second order sliding mode approach. *IFAC Proceedings Volumes*, 39(12):439–444, 2006.

[31] J. N. Franklin. Matrix theory. 1968.

[32] Y. Gao. *Model predictive control for autonomous and semiautonomous vehicles*. PhD thesis, UC Berkeley, 2014.

[33] S. Giannini, A. Petitti, D. Di Paola, and A. Rizzo. Asynchronous max-consensus protocol with time delays: Convergence results and applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(2):256–264, 2016.

[34] M. Goldenbaum and S. Stanczak. Robust analog function computation via wireless multiple-access channels. *IEEE Transactions on Communications*, 61(9):3863–3877, 2013.

[35] M. Goldenbaum, H. Boche, and S. Stańczak. Nomographic gossiping for f-consensus. In *2012 10th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, pages 130–137. IEEE, 2012.

[36] M. Goldenbaum, H. Boche, and S. Stańczak. Harnessing interference for analog function computation in wireless sensor networks. *IEEE Transactions on Signal Processing*, 61(20):4893–4906, 2013.

[37] R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge university press, 2012.

[38] R. Hult, M. Zanon, S. Gros, and P. Falcone. Primal decomposition of the optimal coordination of vehicles at traffic intersections. In *Conference on Decision and Control*, pages 2567–2573, 2016.

[39] R. Hult, M. Zanon, S. Gros, and P. Falcone. Primal decomposition of the optimal coordination of vehicles at traffic intersections. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 2567–2573, 2016.

[40] A. Jadbabaie, J. Lin, and A. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[41] A. Katriniok, P. Sopasakis, M. Schuurmans, and P. Patrinos. Nonlinear Model Predictive Control for Distributed Motion Planning in Road Intersections Using PANOC. In *IEEE Conference on Decision and Control*, 2019.

[42] R. Kianfar, P. Falcone, and J. Fredriksson. A receding horizon approach to string stable cooperative adaptive cruise control. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 734–739. IEEE, 2011.

[43] B.-Y. Kim and H.-S. Ahn. Distributed coordination and control for a freeway traffic network using consensus algorithms. *IEEE Systems Journal*, 10(1):162–168, 2014.

[44] D. B. Kingston and R. W. Beard. Discrete-time average-consensus under switching network topologies. In *2006 American Control Conference*, pages 6–pp. IEEE, 2006.

[45] A. Kortke, M. Goldenbaum, and S. Stańczak. Analog computation over the wireless channel: A proof of concept. In *SENSORS, 2014 IEEE*, pages 1224–1227. IEEE, 2014.

[46] M. Lalo and R. Scattolini. *Advanced and multivariable control*. Pitagoga edirice Bologna, 2014.

[47] P. Lancaster and H. Farahat. Norms on direct sums and tensor products. *mathematics of computation*, 26(118):401–414, 1972.

[48] P. Lancaster and M. Tismenetsky. *The theory of matrices: with applications*. Elsevier, 1985.

[49] H. K. Lee and B. J. Kim. Dissolution of traffic jam via additional local interactions. *Physica A: Statistical Mechanics and its Applications*, 390(23-24):4555–4561, 2011.

[50] S. Limmer, S. Stańczak, M. Goldenbaum, and R. L. Cavalcante. Exploiting interference for efficient distributed computation in cluster-based wireless sensor networks. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 933–936. IEEE, 2013.

[51] J. Liu, A. S. Morse, A. Nedić, and T. Başar. Exponential convergence of a distributed algorithm for solving linear algebraic equations. *Automatica*, 83:37–46, 2017.

[52] P. Liu, U. Ozguner, and Y. Zhang. Distributed mpc for cooperative highway driving and energy-economy validation via microscopic simulations. *Transportation Research Part C: Emerging Technologies*, 77:80–95, 2017.

[53] W. Liu and X. Zang. Over-the-air computation systems: Optimization, analysis and scaling laws. *arXiv preprint arXiv:1909.00329*, 2019.

[54] J. Löfberg. Oops! i cannot do it again: Testing for recursive feasibility in mpc. *Automatica*, 48(3):550–555, 2012.

[55] A. Mahmood and Y. Kim. Leader-following formation control of quadcopters with heading synchronization. *Aerospace Science and Technology*, 47:68–74, 2015.

[56] W. Michiels, C.-I. Morărescu, and S.-I. Niculescu. Consensus problems with distributed delays, with application to traffic flow models. *SIAM Journal on Control and Optimization*, 48(1): 77–101, 2009.

[57] Y. Mo and B. Sinopoli. Communication complexity and energy efficient consensus algorithm. *IFAC Proceedings Volumes*, 43(19):209–214, 2010.

[58] F. Mohseni, E. Frisk, J. Åslund, and L. Nielsen. Distributed model predictive control for highway maneuvers. *IFAC-PapersOnLine*, 50(1):8531–8536, 2017.

[59] F. Molinari and J. Raisch. Automation of road intersections using consensus-based auction algorithms. In *2018 Annual American Control Conference (ACC)*, pages 5994–6001. IEEE, 2018.

[60] F. Molinari and J. Raisch. Efficient consensus-based formation control with discrete-time broadcast updates. *Conference on Decision and Control (CDC) 2019*, 2019.

[61] F. Molinari and J. Raisch. Exploiting wireless interference for distributively solving linear equations. *IFAC-PapersOnLine*, 53 (2):2999–3006, 2020.

[62] F. Molinari, N. N. Anh, and L. Del Re. Efficient mixed integer programming for autonomous overtaking. In *American Control Conference (ACC)*, pages 2303–2308. IEEE, 2017.

[63] F. Molinari, S. Stanczak, and J. Raisch. Exploiting the superposition property of wireless communication for average consensus problems in multi-agent systems. In *2018 European Control Conference (ECC)*, pages 1766–1772. IEEE, 2018.

[64] F. Molinari, S. Stańczak, and J. Raisch. Exploiting the superposition property of wireless communication for max-consensus problems in multi-agent systems. *IFAC-PapersOnLine*, 51(23): 176–181, 2018.

[65] F. Molinari, A. M. Dethof, and J. Raisch. Traffic automation in urban road networks using consensus-based auction algorithms for road intersections. In *2019 18th European Control Conference (ECC)*, pages 3008–3015. IEEE, 2019.

[66] F. Molinari, A. Grapentin, A. Charalampidis, and J. Raisch. Automating lane changes and collision avoidance on highways via distributed agreement. *at-Automatisierungstechnik*, 67(12): 1047–1057, 2019.

[67] F. Molinari, A. Katriniok, and J. Raisch. Real-time distributed automation of road intersections. *IFAC-PapersOnLine*, 53(2): 2606–2613, 2020.

[68] F. Molinari, N. Agrawal, S. Stanczak, and J. Raisch. Max-consensus over fading wireless channels. *IEEE Transactions on Control of Network Systems*, 2021.

[69] J. Monteil, R. Billot, J. Sau, F. Armetta, S. Hassas, and N.-E. E. Faouzi. Cooperative highway traffic: Multiagent modeling and robustness assessment of local perturbations. *Transportation research record*, 2391(1):1–10, 2013.

[70] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on automatic control*, 50(2):169–182, 2005.

[71] S. Mou and A. S. Morse. A fixed-neighbor, distributed algorithm for solving a linear algebraic equation. In *2013 European Control Conference (ECC)*, pages 2269–2273. IEEE, 2013.

[72] S. Mou, J. Liu, and A. S. Morse. A distributed algorithm for solving a linear algebraic equation. *IEEE Transactions on Automatic Control*, 60(11):2863–2878, 2015.

[73] S. Mou, Z. Lin, L. Wang, D. Fullmer, and A. S. Morse. A distributed algorithm for efficiently solving linear equations and its applications (special issue jcw). *Systems & Control Letters*, 91:21–27, 2016.

[74] R. M. Murray. *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[75] B. M. Nejad, S. A. Attia, and J. Raisch. Max-consensus in a max-plus algebraic setting: The case of fixed communication topologies. In *2009 XXII International Symposium on Information, Communication and Automation Technologies*, pages 1–7. IEEE, 2009.

[76] B. M. Nejad, S. A. Attia, and J. Raisch. Max-consensus in a max-plus algebraic setting: The case of switching communication topologies. *IFAC Proceedings Volumes*, 43(12):173–180, 2010.

[77] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, second edition, 2006.

[78] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on automatic control*, 49(9):1520–1533, 2004.

[79] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[80] A. Olshevsky and J. N. Tsitsiklis. Convergence speed in distributed consensus and averaging. *SIAM Journal on Control and Optimization*, 48(1):33–55, 2009.

[81] S. Öncü, J. Ploeg, N. Van de Wouw, and H. Nijmeijer. Cooperative adaptive cruise control: Network-aware analysis of string stability. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1527–1537, 2014.

[82] G. Oriolo, A. De Luca, and M. Vendittelli. Wmr control via dynamic feedback linearization: design, implementation, and experimental validation. *IEEE Transactions on control systems technology*, 10(6):835–852, 2002.

[83] R. Pepy, A. Lambert, and H. Mounier. Path planning using a dynamic vehicle model. In *Information and Communication Technologies, 2006. ICTTA'06. 2nd*, volume 1, pages 781–786. IEEE, 2006.

[84] V. Y. Protasov. The joint spectral radius and invariant sets of the several linear operators. *Fundamentalnaya i prikladnaya matematika*, 2(1):205–231, 1996.

[85] M. Rafiee and A. M. Bayen. Optimal network topology design in multi-agent systems for efficient average consensus. In *49th IEEE Conference on Decision and Control (CDC)*, pages 3877–3883. IEEE, 2010.

[86] R. Rajamani. *Vehicle Dynamics and Control*, volume 2. Springer, 2012.

[87] R. Reghelin and L. V. R. de Arruda. A centralized traffic controller for intelligent vehicles in a segment of a multilane highway. In *IEEE Intelligent Vehicles Symposium*, pages 135–140. IEEE, 2012.

[88] W. Ren and R. W. Beard. Consensus of information under dynamically changing interaction topologies. In *Proceedings of the 2004 American control conference*, volume 6, pages 4939–4944. IEEE, 2004.

[89] W. Ren and R. W. Beard. *Distributed consensus in multi-vehicle cooperative control*. Springer, 2008.

[90] W. Ren and Y. Cao. *Distributed coordination of multi-agent networks: emergent problems, models, and issues*. Springer Science & Business Media, 2010.

[91] W. Ren, R. W. Beard, and T. W. McLain. Coordination variables and consensus building in multiple vehicle systems. In *Cooperative control*, pages 171–188. Springer, 2005.

[92] W. Ren, R. W. Beard, and E. M. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control systems magazine*, 27(2):71–82, 2007.

[93] Y. Ruan and S. K. Jayaweera. Leader-following consensus in vehicle platoons with an inter-vehicle communication network. In *2014 8th International Conference on Telecommunication Systems Services and Applications (TSSA)*, pages 1–6. IEEE, 2014.

[94] R. O. Saber and R. M. Murray. Agreement problems in networks with directed graphs and switching topology. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, volume 4, pages 4126–4132. IEEE, 2003.

[95] C. Samson and K. Ait-Abderrahim. Feedback control of a nonholonomic wheeled cart in cartesian space. In *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pages 1136–1141. IEEE, 1991.

[96] A. Sathya, P. Sopasakis, R. V. Parys, A. Themelis, G. Pipeleers, and P. Patrinos. Embedded nonlinear model predictive control for obstacle avoidance using PANOC. In *ECC*, pages 1523–1528, 2018.

[97] H. Schneider. The concepts of irreducibility and full indecomposability of a matrix in the works of frobenius, könig and markov. *Linear Algebra and its applications*, 18(2):139–162, 1977.

[98] Š. Schwarz. New kinds of theorems on non-negative matrices. *Czechoslovak Mathematical Journal*, 16(2):285–295, 1966.

[99] E. Seneta. *Non-negative matrices and Markov chains*. Springer Science & Business Media, 2006.

[100] J.-y. Shao. Products of irreducible matrices. *Linear algebra and its applications*, 68:131–143, 1985.

[101] G. Shi, B. D. Anderson, and U. Helmke. Network flows that solve linear equations. *IEEE Transactions on Automatic Control*, 62(6):2659–2674, 2016.

[102] L. E. Spence, A. J. Insel, and S. H. Friedberg. *Elementary linear algebra*. Prentice Hall, 2000.

[103] S. Stańczak, M. Goldenbaum, R. L. Cavalcante, and F. Penna. On in-network computation via wireless multiple-access channels with applications. In *2012 International Symposium on Wireless Communication Systems (ISWCS)*, pages 276–280. IEEE, 2012.

[104] L. Stella, A. Themelis, P. Sopasakis, and P. Patrinos. A simple and efficient algorithm for nonlinear model predictive control. In *Conference on Decision and Control*, pages 1939–1944, 2017.

[105] A. Syed, G. G. Yin, A. Pandya, H. Zhang, et al. Control of vehicle platoons for highway safety and efficient utility: Consensus with communications and vehicle dynamics. *Journal of systems science and complexity*, 27(4):605–631, 2014.

[106] I. Trpevski, A. Stanoev, A. Koseska, and L. Kocarev. Discrete-time distributed consensus on multiplex networks. *New Journal of Physics*, 16(11):113063, 2014.

[107] W. Utschick. *Communications in Interference Limited Networks*. Springer, 2016.

[108] M. Vidyasagar. *Nonlinear systems analysis*, volume 42. Siam, 2002.

[109] J. Walrand and P. P. Varaiya. *High-performance communication networks*. Morgan Kaufmann, 2000.

[110] C. Wang, K. Ren, J. Wang, and Q. Wang. Harnessing the cloud for securely outsourcing large-scale systems of linear equations. *IEEE Transactions on Parallel and Distributed Systems*, 24 (6):1172–1181, 2012.

[111] Z. Wang, G. Wu, and M. Barth. Distributed consensus-based cooperative highway on-ramp merging using v2x communications. Technical report, SAE Technical Paper, 2018.

[112] W. L. Winston, M. Venkataramanan, and J. B. Goldberg. *Introduction to mathematical programming*, volume 1. Thomson/Brooks/Cole Duxbury; Pacific Grove, CA, 2003.

[113] J. Wolfowitz. Products of indecomposable, aperiodic, stochastic matrices. *Proceedings of the American Mathematical Society*, 14(5): 733–737, 1963.

[114] P. Xie, K. You, and C. Wu. How to stop consensus algorithms, locally? In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 4544–4549. IEEE, 2017.

[115] X. Zang, W. Liu, Y. Li, and B. Vucetic. Over-the-air computation systems: Optimal design with sum-power constraint. *arXiv preprint arXiv:2005.09139*, 2020.

[116] J. C. Zegers, E. Semsar-Kazerooni, J. Ploeg, N. van de Wouw, and H. Nijmeijer. Consensus control for vehicular platooning with velocity constraints. *IEEE Transactions on Control Systems Technology*, 26(5):1592–1605, 2017.

[117] M. Zheng, M. Goldenbaum, S. Stańczak, and H. Yu. Fast average consensus in clustered wireless sensor networks by superposition gossiping. In *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1982–1987. IEEE, 2012.

[118] V. Zworykin, L. Flory, L. Ress, et al. Electronic control of motor vehicles on the highway. In *Highway Research Board Proceedings*, volume 37, 1958.