Università degli Studi di Cagliari

PH.D. DEGREE

in Electronic and Computer Engineering

Cycle XXXIV

Coordination of Open Multi-Agent Systems

Scientific Disciplinary Sector

ING-INF/04 – Automatica

Ph.D. Student: Zohreh Al Zahra Sanai Dashti

Supervisors Prof. Carla Seatzu
Prof. Mauro Franceschelli

Final exam: Academic Year 2020/2021

Thesis defence: April 2022 Session

Abstract

During the last decade, the problem of consensus in Multi-Agent Systems (MASs) has been studied with special emphasis on graph theoretical methods. Consensus can be regarded as a control objective in which all the agents in a network converge to (or agree "upon") a common value. This is achieved through a given control strategy usually referred to as consensus algorithm. The consensus problem in MASs becomes more challenging in the presence of agents that can join and leave the network, hence the so-called "open" multi-agent systems. This topic is recent in the scientific literature and quickly gaining attention. Thus, this dissertation is motivated by this topic, focusing on the consensus problem on the median consensus value and the average consensus in open multi-agent systems.

Hence, this thesis is first devoted to designing a novel distributed open average consensus protocol for multi-agent systems. The distributed algorithm tracks the average of the agents' state despite the time-varying size and composition of the network. The research activity consists of the design and formal characterization of the convergence properties of the algorithm. The results have been corroborated by numerical simulations.

As for the second part of the research activity, the characterization of the convergence properties of a distributed algorithm to compute and track the median value of a set of numbers in open multi-agent systems has been proposed. A continuous time formulation is considered where the state variables of the agents track with zero error the median value of a set of time-varying reference signals given as input to the agents in a time-varying, undirected network topology. The performance of the proposed protocol is considered in the framework of open multi-agent systems by proposing join and leave mechanisms, i.e., the scenario where agents may join and leave the network during the protocol execution. One notable feature of consensus on the median value is the robustness of the median value, as opposed to the average value, with respect to abnormal or outlier state values. Non-smooth Lyapunov theory is employed to provide convergence guarantees and simple tuning rules to adjust the algorithm parameters.

Apart from studying open multi-agent systems, this dissertation also proposes a distributed scheme for transforming any connected interaction graph with a possible non-

integer average degree into a connected approximately random k-regular graph which is independent of the degree of the initial graph. in more details, we define some local graph transformation rules (consisting of rules for cutting, adding, and moving edges) and provide a distributed implementation. In the resulting process, a random regular graph is obtained while the agents observe and modify only the local structure of the network. As such, the network achieves high expansion ratios and algebraic connectivity, which provide robustness to various structural and functional perturbations.

Acknowledgement

I would like to thank all the people whom supported me during the last three years.

First of all, I wish to show my deep appreciation to my supervisor Prof. **Carla Seatzu** for the continuous support of Ph.D program and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me during all the time of research and writing of this thesis. Thanks a lot, Prof. Seatzu.

Secondly, I would like to express my deep and sincere gratitude to my co-supervisor Prof. **Mauro Franceschelli**. You provided me everything needed to start a Ph.D at Automatic Control Group, Cagliari University. You are aware that, without your help, probably, I would not have started and completed research on this topic. Thanks a lot, Prof. Franceschelli.

Besides my advisors, I would like to thank my thesis Besides my advisors, I would like to thank my thesis evaluators: Prof. **Christoforos Hadjicostis** and Prof. **Paolo Frasca**, for their encouragement and insightful comments.

Finally, I want to appreciate the support and love of my family: my parents and my husband for supporting me throughout my life.

Zohreh Al Zahra Sanai Dashti

List of abbreviations

Below, the list of abbreviations that has been used throughout this thesis, can be found. This list is made per chapter to show where each abbreviation appears first. No new entry will be made if a certain abbreviation returns in a later chapter.

Chapter 1

MAS	Multi-Agent System
OMAS	Open Multi-Agent System
DOAC	Distributed Open Average Consensus

Chapter 2

Chapter 3

OMAS	Open Multi-Agent System
OACP	Open Average Consensus Protocol

Chapter 4

Chapter 5

List of symbols

Below, the list of symbols that has been used throughout this thesis, can be found. This list is made per chapter to easily find the meaning for each symbol.

Standard Symbols	
$\mathbb{R}$	Set of real numbers
$\mathbb{N}$	Set of natural numbers
$\mathbf{1}$	Column vector
$\|.\|$	Cardinality of a generic set
$\|\|.\|\|_2$	Euclidean norm of a vector
$V(t)$	Time-varying set of agents
$G(t)$	Connected time-varying undirected graph
$E(t)$	Time-varying set of edges
$N_i(t)$	Set of neighbors of agent i
$n(t)$	Number of agents
Chapter 2	
$c(t)$	Time-varying consensus function
Chapter 3	
$x_i(t)$	Agent's state at time t
t_{i0}	Instant of time at which agent i joins the network
$u_i(t)$	Control protocol
$C(t)$	Average of the initial state of the agents
$R(t)$	Set of remaining nodes
$D(t)$	Set of departing nodes
$A(t)$	Set of arriving nodes
$E_A(t)$	Set of edges added to the graph by arriving agents
$E_D(t)$	Set of edges removed from the graph by departing agents
$w_{ij}(t)$	Metropolis weights
z_{ij}	Auxiliary state variable
$\bar{x}_i$	State value of the agent at the instant of joining the network
$e_i(t)$	Disagreement value
$\mathscr{V}(t)$	Lyapunov fuction

Chapter 4

$x_i(t)$	Agent's state at time t
$u_i(t)$	Local reference signal given as input
λ	Tunable protocol constant gain
α	Tunable protocol constant gain
$\mathscr{U}(t)$	Set of all entries of vector $u(t)$
$m(u(t))$	Median value
$\mathscr{M}(u(t))$	Median interval
$L_M(t)$	Length of the median interval
$\mathscr{V}_1$	Lyapunov fuction
$\mathscr{V}_2$	Lyapunov fuction
Π	Bounded derivative of all reference signals
$\Delta\tau$	Minimum dwell time separating two consecutive instants

Chapter 5

V	Set of agents t
L	Laplacian matrix
$N_i^1(t)$	Set of 1-hop neighbors of agent i
$N_i^2(t)$	Set of 2-hop neighbors of agent i
$d_i(t)$	Degree of agent i
$d_{\max}(G(t))$	Maximum degree of graph $G(t)$
$d_{\min}(G(t))$	Minimum degree of graph $G(t)$
$\bar{d}(G(t))$	Average degree of graph $G(t)$
$f(G(t))$	Degree range of graph $G(t)$
k	Desired value
$\lambda_2(G(t))$	Second-smallest eigenvalue of the graph Laplacian (lgebraic connectivity)
$\bar{\mathbb{G}}$	Set of regular graphs whose the degree of all nodes is equal to k
$\tilde{\mathbb{G}}$	Set of non-regular graphs

Contents

Chapter 1

Introduction

1.1 Motivation

A Multi-Agent System (MAS) is a large set of dynamical systems which interact within a network. Many real systems in nature and human society can be modeled as multiagent systems, thus in the last two decades cooperative systems have received compelling attention in different research fields such as physics sciences, mathematics, economic science, and engineering. One of the most heavily investigated problems in multi-agent systems is the consensus problem, i.e., the design of a distributed control strategy to drive the state variables of each agent to the same value. Inspired by the natural occurrence of flocking and formation, many researchers have focused their work on the synchronization, consensus, and coordination of dynamic multi-agent systems [1, 2]. When it comes to examples in engineering, MASs are also used to model several networks of systems such as multi-robot systems [3,4], sensor networks [5,6], networks of energy storage systems [7], and so on which interact over a communication or sensing network. One of the most fundamental features of these examples is that agents can login (join) and logout (leave) the network. However, in the literature on multi-agent systems only few papers have addressed this scenario.

In the past decade, the consensus problem has been investigated from several different perspectives. Most early papers on multi-agent systems address consensus and synchronization problem without considering the presence of a leader node, so all nodes are commanded to converge toward a not prescribed common evolution. Consensus protocols which converge to the minimum, maximum, median and average value [8–10] are examples of this method. Synchronization, in the sense of cooperative tracking, has been then studied by adding a leader, the so-called leader-following consensus that imposes the desired behaviour to a group of agents to achieve the command trajectory (e.g., see [11, 12]. Other perspectives on consensus problems include finite-time consensus, where the state variables of networked multi-agent systems are said to reach the desired value if there exists a finite convergence rate. Finite-time consensus has been studied in [10, 13, 14], just to mention only a very small sample of this huge literature. In [8] the problem of exact average consensus over time-varying digraphs was investigated by employing an additional variable to keep track of the state changes of each agent. The protocol in [9] guarantees perfect average consensus despite a variety of challenging scenarios, including possible packet drops in the communication links, and imprecise knowledge of the network. In [10], consensus on the median value is achieved in finite time under static communication graphs and it is shown that the considered algorithm is resilient with respect to both outliers and uncooperative agents if a graph theoretical condition based on k-connectivity is satisfied.

Most existing literature on multi-agent systems has studied the consensus problem by considering networks where the set of agents does not change. Thus, motivated in this thesis, we firstly design a novel discrete-time distributed open average consensus protocol to let each agent track the time-varying value of the average initial state of the agents in a scenario where agents can join and leave the network during the protocol execution. Then, we focus on the problem of dynamic consensus of a network of agents corresponding to continuous-time systems wherein the state variables of the agents track with zero error the median value of a set of time-varying reference signals given as input to the agents under a time-varying, undirected network topology. Furthermore, we analyse the performance of the protocol in the framework of open multi-agent systems by considering join and leave events, i.e., the scenario where agents may join and leave the network during the protocol execution.

1.2 Related literature

A recent development in control and systems theory considers open-multi-agent systems as networks of coupled dynamical systems that can login and logout from the network, thus change the number of state variables during the transient and steady-state behavior of the multi-agent system. Recent works of the still limited literature on the topic can be found in [15–23]. In particular, the authors in [15] dealt with a class of discrete-time dynamic average consensus algorithms wherein a group of agents can track the average of their reference inputs. They showed the proposed algorithm is robust to the dynamic change of communication topologies as well as the joining and leaving (or failure) of nodes, albeit, stability analysis was not performed on the open scenario. A discrete time average consensus algorithm for open multi-agent systems has been analyzed in terms of three scale-independent quantities under the assumption that departure and arrival occurred at pre-determined times [16]. Successively, the authors have extended [16] by considering agents that can login and logout at random times, wherein the network size was fixed, i.e., each departure of an agent is instantaneously followed by an arrival [17].

This issue has then been investigated in [18], where the main idea is to achieve consensus on the maximum value of the state of the agents by using randomized gossip interactions between them. Additionally, novel definitions of stability along with an open proportional dynamic consensus algorithm have been reported for open multi-agent systems in [19] and [20] wherein agents estimate the time-varying average of a set of reference signals. Another consensus problem of discrete-time open multi-agent systems with stochastic interactions controlled by a Bernoulli process is studied in [21]. Finally, decentralized optimization algorithm in this setting is receiving attention, such as in [22] where time-varying objective functions are considered, and in [23] where the stability of decentralized gradient descent is analyzed and it is shown that decentralized gradient descent is stable when agents/functions change over time if their objectives are sufficiently smooth.

1.3 Possible applications

MAS is applied to several real systems networks such as multi-robot systems, sensor networks, energy storage systems, etc., interact with their environment and accomplish actions to reach their goals. Some of these real applications can be found in [4–7, 24]. The authors in [4] formulated a general distributed coordination problem in multi-robot systems. In [7, 24] the problem of voltage and frequency restoration in microgrid systems is addressed in a multi-agent fashion where the generators' local controllers play the role of cooperative agents communicating over a network. The qualitative analysis of Java-based multi-agent platforms for wireless sensor networks (WSN) proposed in [5, 6] to facilitate the decision for choosing a given agent platform to develop and optimize agent-based WSN applications. Several other real-world applications can be described with the multi-agent system framework. It is worth noting that one of the most significant features of these applications is that agents can join and leave the network. Therefore, the consensus algorithms proposed in this dissertation can be implemented in these real applications.

1.4 Structure and topics of the thesis

We organise this dissertation into six chapters. In **Chapter 1**, we firstly present the introduction of the multi-agent systems to the reader, which is useful to describe different real-world scenarios. Then, we provide the motivation and the related literature. Lastly, we include a brief outline of the research activities in this thesis. We outline the mathematical background and useful properties from other authors for later use in **Chapter 2**. The rest of this dissertation covers the following main topics

In **Chapter 3**, we study the problem of discrete-time average consensus in Open Multi-Agent Systems (OMAS), where agents may join and leave the network at any time. Distributed algorithms to achieve average consensus in networks have been widely investigated [25–31]. For average consensus on static (i.e., time-invariant, fixed) digraphs, [27, 30] justify that a balanced and strongly connected topology is necessary and sufficient to guarantee convergence. Weight-balanced digraphs are essential in distributed averaging. The Authors in [32] propose distributed algorithms over static topologies to solve the weight balancing problem when the weights were arbitrary non-negative real numbers. Given a general strongly connected digraph, [33], considers how to find corresponding weight-balanced digraphs, and proposes two algorithms to achieve this goal by selecting the out-edge weights to balance the in-degrees and out-degrees. The authors in [34] design an asynchronous broadcasted gossip algorithm to calculate the (possibly weighted) average of the initial measurements of the nodes in the network and show that the broadcast gossip algorithm converges almost surely to a consensus value, which is in expectation equal to the desired average of initial measurements. A recent overview on event-triggered average consensus and control of multiagent systems can be referred to [35]. It should be pointed out that most of the existing literature on average consensus works considers static networks where their composition remains unchanged throughout the whole process. Hence, in **Chapter 3** we focus on the problem of discrete-time average consensus in open multi-agent systems, where agents may join and leave the network at any time. To address this problem, we propose a novel Distributed Open Average Consensus Protocol (DOACP) to track with a bounded error the time-varying value of the average of the agents' state. We also provide a novel convergence analysis to address a time-varying number of state variables. Finally, we present numerical simulations to corroborate the theoretical analysis.

In **Chapter 4**, we focus on Dynamic Consensus on the Median Value in Open Multi-Agent Systems. Although the literature on the open multi-agent systems topic is limited, the literature on dynamic consensus is rich. One of the early works on the topic is [36], which uses Laplacian feedback and provides as inputs to the agents the derivative of a set of reference signals. A high order continuous-time dynamic average consensus protocol is then reported in [37]. The requirement on precise initialization of the algorithm makes it not robust to initialization errors or changes in the network composition. To guarantee robustness with respect to initial errors, [25, 38, 39]propose alternative dynamic average consensus algorithms whose convergence is guaranteed that the graph representing the network topology is connected, static and balanced. Other notable mentions in the consensus literature are [40–42] (the so called ratio consensus) to solve consensus on unbalanced digraphs by exploiting the ratio of two state variables that perform linear state updates with appropriate, distinct, initialization. The proposed method in this chapter is based on a discontinuous local interaction rule. We refer to [43–45] for an exhaustive tutorial on how to study discontinuous gradient flows and discontinuous feedback systems by means of non-smooth Lyapunov theory.

The algorithm we present converges in finite time. Approaches that obtain consensus in finite time can be found in [46–52]. In particular, in [46],[47] and [48] the propose approaches are capable of tracking with zero error the average value of time-varying smooth reference signals with bounded derivatives, in finite time on static undirected graph topologies. The dynamic average consensus protocol in [46] gurantees perfect tracking in finite time using one-hop communication, but it suffers from the limitation of not being robust to initial conditions. The protocol in [47] converges to zero error in finite time and is also robust to initial conditions, however it requires two-hop communication. The protocol in [48] uses one-hop communication, but the error synchronizes to zero exponentially, rather than in finite time. In [53] using continuous local interactions, finite time consensus is studied for single integrators and second order systems with unknown non-linear dynamics. It is worth noting that, most averaging approaches in large scale networks suffer from a considerable limitation: the presence of even a single outlier value within the considered data may affect the network behavior. This is a strong motivation to pursue consensus strategies on robust statistics such as the median as opposed to the average. In [54] and [55], to overcome the drawbacks of average consensus, the authors proposed algorithms to solve the consensus on the median value of a set of numbers. In more detail, in [55], consensus on the median value is achieved in finite time under static communication graphs and it is shown that the considered algorithm is resilient with respect to both outliers and uncooperative agents if a graph theortical condition based on k-connectivity is satisfied. In [54], a robust distributed consensus on the median value is proposed in the presence of matching perturbations to the agents dynamics. In contrast to this paper, in **Chapter 4** we consider a dynamic consensus problem in continuous time where the state variables of the agents track with zero error the median value of a set of time-varying reference signals given as input to the agents in a time-varying, undirected network topology. Then, we consider the performance of the protocol in the framework of open multi-agent systems by proposing join and leave mechanisms, i.e., scenarios where agents may join and leave the network during the protocol execution. We characterize the finite-time convergence properties and tracking error of the considered protocol in the case of inputs with bounded variations. One notable feature of consensus on the median value is the robustness of the median, as opposed to the average, with respect to abnormal or outlier values of inputs which represent the outcome of a measurement or estimation process, thus significantly increasing the robustness of the estimation for large scale networks. We use non-smooth Lyapunov theory to provide convergence guarantees and simple tuning rules to adjust the algorithm parameters.

In addition to studying the topic of open multi-agent systems, this dissertation addresses the problem of robustness improvement of multi-agent systems against perturbations such as failures, noise, or malicious attacks. The robustness of a multi-agent network to perturbations strongly depends on the structure of the corresponding graph, thus in recent years, graph-theoretic analysis of networked systems has attracted interest many researchers. One of the basic concepts of graph theory in mathematics and computer science is connectivity, which is a noteworthy measure of its resilience as a network. A graph calls a Q-connected graph if node connectivity or edge-connectivity be greater or equal to Q. Indeed, Q is the minimum number of nodes or edges that need to be cut in order to transform a connected graph into two or more components. Generally, high-connectivity graphs are more robust against their component failure. Connectivity can also be characterized by the second-smallest eigenvalue of the Laplacian matrix of graph that is named the algebraic connectivity of the graph. The magnitude of this value reflects how well connected the overall graph is, and has been used in analysing the robustness and synchronizability of networks. If the algebraic connectivity of a graph is small, it is more likely to be disconnected by removing only a few edges (or nodes).

The connectivity of a graph can be increased by adding more edges to the graph. However, adding more edges to a graph is not practically appropriate. Since each edge is considered to reflect some communications, sensing, or a physical link between the corresponding nodes, then more edges entail more resources. In addition, increasing many edges may lead to higher vulnerability to the cascading failures such as epidemics. Therefore, in real-world systems, having a small number of edges (i.e., sparsity) is important. Hence, researchers were motived to present a decentralized protocol for transforming any connected interaction graph into a connected random regular graph [56–58]. A regular graph is a graph where is well connected yet sparse whose each node has the same number of neighbors and a regular graph with nodes of degree k is called a k-regular graph or regular graph of degree k. Furthermore, selecting a graph uniformly at random from the set of all k-regular graphs is entitled to a random k-regular graph. In more detail, in [56] and [57], to robustify multi-agent systems against perturbations, the authors proposed algorithms where any graph with an integer average degree, $m \in \mathbb{N}$, was transformed into a random k-regular graph while the protocol in [58], rewires any graph with a possible non-integer average degree of m into a random k-regular graph for some $k \in [m, m+2]$.

It is worth noting that, k-regular random graph built by the aforementioned works is dependent on the average degree of the initial graph. Thus, in **Chapter 5**, we design a distributed scheme for transforming any connected interaction graph with a possible non-integer average degree into a connected approximately random k-regular graph which is independent of the degree of the initial graph. Hence, the network is well connected with a relatively small number of links that leads to improved robustness of networks. Finally, we provide numerical simulations to demonstrate the effectiveness of the proposed solution.

Chapter 2

Mathematical Background

This chapter is structured as follows. In Section 2.1, we give some basics on notation. After that, in Section 2.2, we illustrate preliminaries of graph theory in open MASs. We provide multi-agent modeling of dynamical systems in Section 2.3. Finally, in Section Section 2.4 for later use we present some useful properties of other literature

2.1 Nomenclature

Let $\mathbb{R}$ and $\mathbb{N}$ express, respectively, the sets of real and positive integer numbers. Given a matrix $A \in \mathbb{R}^{n\times n}$, its transpose is denoted by A^T. The generic element (i, j) of matrix A is denoted as a_{ij}. A matrix A is said to be a nonegative matrix if all its elements satisfy $a_{ij} \geq 0$. A nonnegative matrix is called row-stochastic if $A\mathbf{1} = \mathbf{1}$ where $\mathbf{1}$ is a column vector with all unitary elements of appropriate dimension. If the matrix is also column-stochastic, i.e., $\mathbf{1}^T A = \mathbf{1}^T$ then matrix A is called a doubly stochastic matrix. In this thesis we denote the cardinality of a generic set S as $|S|$ and the Euclidean norm of a vector S as $||S||_2$.

2.2 Network topology in open MAS

Let $G(t) = (V(t), E(t))$ be a connected time-varying undirected graph, where $t \in \mathbb{N}$ is a time index, $V(t)$ is a time-varying set of integer numbers which represents agents logged in the network at time t and $E(t) \subseteq \{\ V(t) \times V(t)\}$ is the set of edges representing information exchange between agents. Each agent is represented by an integer $i \in V(t)$. There exists an edge between agents i and j if $(i, j) \in E$. Let $N_i(t) = \{j \in V(t) : (i, j) \in E(t)\}$ be the set of neighbors of agent i, i.e., agents that share an edge with agent i. The time-varying degree of each node is defined as the cardinality of its neighborhood, $\Delta_i(t) = |N_i(t)|$ where $|\cdot|$ denotes the cardinality of a set. We denote by $n(t) = |V(t)|$ the number of agents logged in the network at time t. Let us define by $R(t) = V(t) \cap V(t+1)$, $D(t) = V(t) \setminus V(t+1)$ and $A(t) = V(t+1) \setminus V(t)$, respectively, the sets of remaining nodes that belong to both $V(t)$, and $V(t+1)$, departing nodes that belong to $V(t)$ but not to $V(t+1)$ and arriving nodes that belong to $V(t+1)$ but not to $V(t)$. We also define the edge sets $E_A(t) = \{(i,j) : i \in V(t) \setminus D(t), j \in A(t)\}$ representing the edges added to the graph by arriving agents at time t and set $E_D(t) = \{(i,j) : i \in V(t) \setminus D(t), j \in D(t) \text{ or } i,j \in D(t)\}$ representing edges removed from the graph by departing agents.

2.3 Multi-agent modeling of dynamical systems

A dynamical system is a physical system whose states are represented by the points of a set. These states evolve with time according to a deterministic law, specified by a differential or difference equation. For clarity, we present the state-space representation of a dynamical system in the block diagram 2.1, which models a physical system as a set of state variables, ($x(t)$ or ($x(k)$), input, ($u(t)$ or $u(k)$), and output ($y(t)$ or $y(k)$) signals related by first-order differential equations (continuous-time) or difference equations (discrete-time). We use the term dynamical system to refer to either continuous-time or discrete-time dynamical systems.

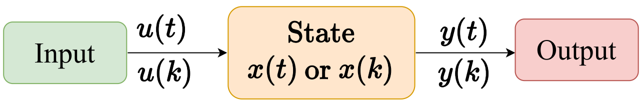

Figure 2.1: State-space representation of a dynamical system.

2.3.1 Discrete-time dynamical system

A discrete dynamical system is a dynamical system whose state evolves over state space in discrete time steps according to a law. A dynamical system evolving in discrete-time is described by:

$$x(k+1) = f\big(x(k), u(k)\big), \quad k \in \mathbb{N} \tag{2.1}$$

where $x(k)$ is the current state of the system, and $x(k+1)$ is the state of the system after one interval of time has passed. By iterating the above calculation multiple times, we can find subsequent states of the system. $f : X \to X$ is a map and the k-th iterate of map f is the n-fold composition $f^n = f \circ \ldots \circ f$, and hence the system can also be modeled by $x(k+1) = f^k\big(x(0)\big)$.

2.3.2 Continuous-time dynamical system

A continuous dynamical system is a dynamical system whose state evolves over the state space continuously over according to a fixed rule. A dynamical system evolving in continuous-time is modeled by:

$$\dot{x}(t) = \Phi\big(x(t), u(t)\big), \quad t \in \mathbb{R} \tag{2.2}$$

where $x(t)$ represents the current state of the system, which identifies the one parameter family of maps $\Phi(t, x(t)) : X \to X$ for $t \in \mathbb{R}$, solutions to the initial value problem.

2.4 Useful properties

For later use, we provide the following useful properties:

Definition 2.4.1: Finite-Time Consensus

The state variables $x_i(t) \in \mathbb{R}$, $i \in V(t)$, of a networked multi-agent system are said to reach *finite-time consensus* if there exists a real value $T \in (0, \infty)$ and a real function $c(t) : \mathbb{R} \to \mathbb{R}$ such that

$$x_i(t) = c(t), \quad \forall i \in V(t), \quad \forall t \geq T. \tag{2.3}$$

Function $c(t)$ is the time-varying consensus function.

Consider the dynamical system

$$\dot{x}(t) = f(x(t)), \qquad x(t) \in \mathbb{R}^n, \qquad x(0) = x_0 \in \mathbb{R}^n, \tag{2.4}$$

where $f(x(t)) : \mathbb{R}^n \to \mathbb{R}^n$, is defined almost everywhere, i.e., it is defined for every $x(t) \in \mathbb{R}^n \setminus W$, where W is a subset of $\mathbb{R}^n$ of measure zero. Furthermore, $f(x(t))$ is measurable in an open region $Q \subset \mathbb{R}^n$ and for all compact sets $D \subset Q$ there exists a constant A_D such that $\|f(x(t))\| \leq A_D$ almost everywhere in D.

If the differential equation (2.4) has discontinuous right-hand side, following [59] we understand the corresponding solution in the so-called *Filippov sense* as the solution of an appropriate differential inclusion, as explained in the next definition.

Definition 2.4.2: Filippov Solution

A vector function $x(\cdot) \in \mathbb{R}^n$ is called a Filippov solution of (2.4) on $[t_0, t_1]$ if $x(\cdot)$ is absolutely continuous on $[t_0, t_1]$ and, for almost all $t \in [t_0, t_1]$, satisfies the differential inclusion

$$\dot{x} \in K(x) \triangleq \bigcap_{\delta > 0} \bigcap_{\mu(N)=0} co\left\{f\left(B(x, \delta) \setminus N, t\right)\right\}, \tag{2.5}$$

where $\bigcap_{\mu(N)=0}$ denotes the intersection over all sets N of Lebesgue measure zero, $co\{\cdot\}$ denotes the convex hull and $B(x, \delta)$ is a ball of radius δ centered at x.

If $f(x)$ is measurable and locally bounded then the corresponding set-valued map is upper semicontinuous, compact, convex valued and locally bounded so that the differential inclusion (2.5) possesses a Filippov solution for each initial condition x_0. The reader is referred to [44] for a comprehensive tutorial on the topic.

We now recall the definition of *Clarke's Generalized Gradient* [60].

Definition 2.4.3: Clarke's Generalized Gradient

Let $\mathscr{V}(x):\mathbb{R}^n \to \mathbb{R}$ be a locally Lipschitz continuous function. Its Clarke's generalized gradient $\partial\mathscr{V}(x)$ is defined as

$$\partial\mathscr{V}(x) \triangleq co\left\{\lim_{i\to\infty} \nabla\mathscr{V}(x_i)|x_i \to x, x_i \notin \Omega_V \cup N\right\},$$

where $\nabla\mathscr{V}$ denotes the conventional gradient, $x_i \in \mathbb{R}^n$ represents a point of an infinite succession which converges to $x \in \mathbb{R}^n$ as i grows to infinity, Ω_V is a set of Lebesgue measure zero which contains all points where $\nabla\mathscr{V}(x)$ does not exist, and N is an arbitrary set of measure zero.

The Clarke's generalized gradient coincides with the standard gradient at the points where the standard derivative of the scalar function exists. Further details and examples of computation can be found in [43, 44, 60]. Next, we recall the definition of set-valued Lie derivative.

Definition 2.4.4: Set-valued Lie derivative

Given a locally Lipschitz function $\mathscr{V}(x)$, where $x \in \mathbb{R}^n$ is governed by the differential inclusion $\dot{x} \in K(x)$, the set-valued Lie derivative of $\mathscr{V}(x)$ at x is

$$\widetilde{\mathscr{L}}\mathscr{V}(x) = \{a \in \mathbb{R} | \exists v \in K(x) \text{ such that } \zeta \cdot v = a, \forall \zeta \in \partial\mathscr{V}(x)\}. \tag{2.6}$$

The set-valued Lie derivative allows the study of the evolution of a Lyapunov function along the Filippov solutions of the dynamical system according to the next theorem.

Theorem 2.4.1: Evolution along Filippov solutions [60]

Let $x(t):[t_0,t_1] \to \mathbb{R}^n$ be a Filippov solution of (2.5). *Let $\mathscr{V}(x(t))$ be a locally Lipschitz and regular function. Then $\frac{d}{dt}(\mathscr{V}(x(t)))$ exists a.e. and $\frac{d}{dt}(\mathscr{V}(x(t))) \in \widetilde{\mathscr{L}}\mathscr{V}(x(t))$ a.e..* ■

A generalization of the extended Lyapunov theorem for non-smooth analysis appropriate for the study of consensus problem is the following ([43],[55]).

Theorem 2.4.2

Let $M = span(\mathbf{1}_n)$ be the subspace spanned by vector $\mathbf{1}_n$. Consider a scalar function $\mathscr{V}(x(t)) : \mathbb{R}^n \to \mathbb{R}$, with $\mathscr{V}(x(t)) = 0$ $\forall x \in M$ and $\mathscr{V}(x(t)) > 0$ $\forall x \notin M$. Let $x(t) : \mathbb{R} \to \mathbb{R}^n$ and $\mathscr{V}(x(t))$ be absolutely continuous on $[t_0, \infty)$ with $\frac{d}{dt}(\mathscr{V}(x(t))) \leq -\varepsilon < 0$ a.e. on $\{t | x(t) \notin M\}$. Then, $\mathscr{V}(x(t))$ converges to 0 in finite time and $x(t)$ reaches the subspace M in finite time as well. ■

As first step in the analysis, it is worth mentioning that since the corresponding right-hand side is uniformly bounded [44], there exists a Filippov solution to (4.1) for every initial condition of (4.1)

Chapter 3

Average Preserving Discrete-time Consensus

3.1 Introduction

In this chapter, we address the problem of discrete-time average consensus in Open Multi-Agent Systems (OMAS), i.e., multi-agent systems that are subject to arrivals and departures of agents. To solve this problem, we propose a novel Distributed Open Average Consensus Protocol (OACP) to track with a bounded error the time-varying value of the average of the agents' state. We present a novel convergence analysis to address a time-varying number of state variables. Lastly, we provide numerical simulations to confirm the theoretical analysis.

The structure of this chapter is as follows: In Section 3.2 we present the main contributions. Then, we detail the statement of the problem in Section 3.3. After that, we discuss the proposed algorithm in Section 3.4. We provide the convergence analysis in Section 3.5. In Section Section 3.6 we carry out numerical simulations. Finally, in Section Section 3.7 we discuss concluding remarks and future research directions.

3.2 Main contributions

The main contribution of this chapter is a novel distributed open average consensus protocol which tracks the average of the agents' initial state despite the time-varying size and composition of the network. We provide a formal convergence analysis and derive bounds on the convergence error during the join and leave process by using an appropriate Lyapunov functional. The results are corroborated by numerical simulations.

3.3 Problem statement

We consider a discrete-time open MAS with topology represented by an undirected graph $G(t) = (V(t), E(t))$ where the agents have dynamics described by:

$$x_i(t+1) = x_i(t) + u_i(t) \quad i \in V(t), \quad x_i(t_{i0}) = \bar{x}_i, \tag{3.1}$$

where $x_i(t) \in \mathbb{R}$ represents the agent's state at time t, t_{i0} is the instant of time at which agent i joins the network and $u_i(t) \in \mathbb{R}$ is a control protocol.

Our objective is to design a discrete-time distributed control protocol u_i for agents as in eq. (3.1) to let each agent track the time-varying value of the average of the initial state of the agents $c(t)$ currently logged in the network at time t, i.e.,

$$c(t) = \frac{1}{n(t)} \sum_{i \in V(t)} \bar{x}_i, \tag{3.2}$$

where the time-varying number of agents $n(t)$ is unknown to the agents themselves. The value of $c(t)$ changes in time due to the departing and arriving agents. Note that the standard discrete-time consensus protocol would be biased in this open scenario because each arriving and departing agent changes permanently the consensus value due to its past influence to the network.

3.4 Proposed open average consensus protocol

In this section, we present the open average consensus protocol (OACP), which is stated in detail in Algorithm 1.

First, let us introduce some preliminary definitions. To each edge (i, j) it is associated a weight w_{ij}, the so-called *Metropolis* weights [61], defined as:

$$w_{ij}(t) = \begin{cases} \dfrac{1}{1+\max\left\{\Delta_i(t), \Delta_j(t)\right\}}, & (i, j) \in E(t) \\ -\sum\limits_{j \in N_i(t)} w_{ij}(t), & i = j \\ 0. & \text{otherwise} \end{cases} \tag{3.3}$$

Let matrix $W(t)$ have elements w_{ij} as in (3.3) and let $P(t) = I + W(t)$. It can be easily verified that matrix $P(t)$ is a doubly stochastic matrix by construction.

We now introduce an auxiliary state variable $z_{ij}(t) \in \mathbb{R}$ for each edge $(i, j) \in E(t)$ and $z_{ji}(t)$ for $(j,i) \in E(t)$ in the network. Therefore, we define as the state of each generic agent i the set of state variables $\left\{x_i, z_{ij} : (i, j) \in N_i\right\}$.

The basic ideas behind Algorithm 1 are as follows: i) When an agent $i \in V(t)$ leaves the network at time t, all its states and edges are removed from the network. Furthermore, all the state variables of its neighbors are updated according to eq. (3.5). All its neighbors j which are not leaving consider their auxiliary state variable $z_{ji}(t)$ (see eq.

Algorithm 1: Open average consensus protocol (OACP)

Initialize:

$$G(0) = (V(0), E(0));$$
$$x_i(0) = \bar{x}_i, \quad \forall i \in V(0);$$
$$z_{ij}(0) = 0, \quad \forall (i,j) \in E(0);$$
$$t = 0.$$

- **At each time t graph $G(t) = (V(t), E(t))$ is updated as follows:**

$$V(t+1) = (V(t) \cup A(t)) \setminus D(t),$$
$$E(t+1) = (E(t) \cup \{(i,j) : i \in A(t)\}) \setminus \{(i,j) : i \in D(t)\}.$$

- **At each time t each agent i executes a local state update with its neighbors $j \in N(t)$ as follows:**

if $\big($*Agent i departs form the network, i.e.,* $i \in D(t)\big)$ **then**
  Remove the agent's state and its edges from the network.
end
if $\big($*Agent i arrives in the network, i.e.,* $i \in A(t)\big)$ **then**
  Update edge weights according to Eq. (3.3), set $x_i(t) = \bar{x}_i$ and $z_{ij}(t) = 0$ for all $j \in N_i(t)$, set

$$x_i(t+1) = x_i(t), \quad \forall i \in A(t). \tag{3.4}$$

end
if $\big($*Agent i remains in the network, i.e.,* $i \in R(t)\big)$ **then**
  Update edge weights according to Eq. (3.3), set

$$x_i(t+1) = x_i(t) + \sum_{j \in N_i(t) \setminus D(t)} w_{ij}(t)\big(x_j(t) - x_i(t)\big) - \sum_{j \in N_i(t) \cap D(t)} z_{ij}(t), \tag{3.5}$$

  and

$$z_{ij}(t+1) = z_{ij}(t) - w_{ij}(x_i(t) - x_j(t)), \quad \forall j \in N_i(t). \tag{3.6}$$

end

(3.6)) and add it to their state, (see eq. (3.5)). By this strategy we are able to remove the past influence on the consensus value of the departing agent from the network. ii) When an agent arrives in the network (thus sharing at least one edge with an agent already connected to the network), according to Algorithm 1, all edge weights are updated in accordance with eq. (3.3) and its state is initialized as follows:

$$x_i(t) = \bar{x}_i \tag{3.7}$$

where $\bar{x}_i \in \mathbb{R}$ is the state value of the agent at the instant of joining the network.

We consider the following two assumptions on the network topology.

Assumption 3.4.1

Graph $G(t)$ is undirected and connected for all $t \geq 0$. ■

Assumption 3.4.2

There exists $n_{max} \in (0, \infty) : n(t) \leq n_{max}, \ \forall t \geq 0.$ ■

Note that we assume that there exists a finite value for n_{max} for analysis purposes, but this value is unknown to the agents.

3.5 Convergence analysis

In this section, we discuss the convergence properties of Algorithm 1. In particular, we show that the state of each agent $i \in V(t)$ tracks the average of $\bar{x}_i$ in the open scenario.

The convergence analysis is presented in three parts.

Part 1: We prove that the sum of all state variables of the agents is equal to the sum of all the initial states $\bar{x}_i$ at any time, i.e, $\sum_{i \in V(t)} x_i(t) = \sum_{i \in V(t)} \bar{x}_i$.

Part 2: We show that the state variables of the agents converge exactly to the average value of the initial state of the agents if no agent joins or leaves.

Part 3: We prove that when agents join or leave the network, the time-varying average of the initial states can be tracked with a bounded error by the MAS with no bias, i.e., if agents stop joining or leaving the network, it converges exactly to the average of the initial states, erasing the influence on the consensus value of the agents who left the network.

- Part 1. The next theorem formally proves the first part of the convergence results, namely that the sum of all state variables in the network at time t is, thanks to Algorithm 1, are always equal to the sum of the initial states of the agents connected to the network at the same instant of time.

Theorem 3.5.1

Consider agents that join or leave the network arbitrarily while executing Algorithm 1. If the time-varying graph $G(t)$ is connected for all $t \geq 0$, then

$$\sum_{i \in V(t)} x_i(t) = \sum_{i \in V(t)} \bar{x}_i, \qquad \forall t \geq 0. \tag{3.8}$$

Proof of Theorem 3.5.1 *Let us compute* $\sum_{i \in V(t+1)} x_i(t+1)$ *by replacing* (3.5) *into it, namely,*

$$\sum_{i \in V(t+1)} x_i(t+1) = \sum_{i \in V(t) \setminus D(t)} \Big\{ x_i(t) + \sum_{j \in N_i(t) \setminus D(t)} w_{ij}(t) \big(x_j(t) - x_i(t) \big) - \sum_{j \in N_i(t) \cap D(t)} z_{ij}(t) \Big\} + \sum_{i \in A(t)} x_i(t). \tag{3.9}$$

According to (3.7) *when an agent joins the network it is initialized as* $x_i(t) = \bar{x}_i$*, thus Eq.* (3.9) *can be recast as:*

$$\sum_{i\in V(t+1)} x_i(t+1) = \sum_{i\in V(t)\setminus D(t)} \Big\{ x_i(t) + \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(x_j(t)-x_i(t)\big) - \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\} + \sum_{i\in A(t)} \bar{x}_i. \tag{3.10}$$

Then, we can rewrite (3.10) *as:*

$$\sum_{i\in V(t+1)} x_i(t+1) = \sum_{i\in V(t)\setminus D(t)} x_i(t) + \sum_{i\in A(t)} \bar{x}_i + \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(x_j(t)-x_i(t)\big) \Big\} - \sum_{i\in V(t)\setminus D(t)} \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t). \tag{3.11}$$

Exploiting the advantage of the symmetry of interactions, we can get:

$$\sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(x_j(t)-x_i(t)\big) \Big\} = 0. \tag{3.12}$$

Thus, according to (3.12)*, Eq.* (3.11) *can be rewritten as:*

$$\sum_{i\in V(t+1)} x_i(t+1) = \sum_{i\in V(t)\setminus D(t)} x_i(t) + \sum_{i\in A(t)} \bar{x}_i - \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\}. \tag{3.13}$$

Let us rewrite the last term of Eq. (3.13) *as:*

$$\sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\} = \sum_{(i,j)\in E_D(t)} z_{ij}(t). \tag{3.14}$$

Then, Eq. (3.13) *can be recast as:*

$$\sum_{i\in V(t+1)} x_i(t+1) = \sum_{i\in V(t)\setminus D(t)} x_i(t) + \sum_{i\in A(t)} \bar{x}_i - \sum_{(i,j)\in E_D(t)} z_{ij}(t). \tag{3.15}$$

Additionally, it can be justified that the summation of auxiliary state variables $z_{ij}(t) \in \mathbb{R}$ *over all edges of departing nodes are equal to the summation of states of the departing nodes, i.e.,*

$$\sum_{(i,j)\in E_D(t)} z_{ij}(t) = -\sum_{i\in D(t)} x_i(t). \tag{3.16}$$

Hence, we can write Eq. (3.15) *as:*

$$\sum_{i\in V(t+1)} x_i(t+1) = \sum_{i\in V(t)\setminus D(t)} x_i(t) + \sum_{i\in A(t)} \bar{x}_i + \sum_{i\in D(t)} x_i(t). \tag{3.17}$$

Finally, Eq. (3.17) *can be recast as:*

$$\sum_{i\in V(t+1)} x_i(t+1) = \sum_{i\in V(t)} x_i(t) + \sum_{i\in A(t)} \bar{x}_i. \tag{3.18}$$

Taking into account $x_i(0) = \bar{x}_i$ *for all* $i \in V(0)$ *and according to Eq.* (3.18)*, it holds:*

$$\begin{aligned}
\sum_{i\in V(1)} x_i(1) &= \sum_{i\in V(0)} x_i(0) + \sum_{i\in A(0)} \bar{x}_i,\\
\sum_{i\in V(1)} x_i(1) &= \sum_{i\in V(0)} \bar{x}_i + \sum_{i\in A(0)} \bar{x}_i = \sum_{i\in V(1)} \bar{x}_i,\\
\sum_{i\in V(2)} x_i(2) &= \sum_{i\in V(1)} \bar{x}_i + \sum_{i\in A(1)} \bar{x}_i = \sum_{i\in V(2)} \bar{x}_i,\\
\sum_{i\in V(t+1)} x_i(t+1) &= \sum_{i\in V(t)} \bar{x}_i + \sum_{i\in A(t)} \bar{x}_i = \sum_{i\in V(t+1)} \bar{x}_i.
\end{aligned}$$

Thus for $t \geq 0$ *we can write the following result*

$$\sum_{i\in V(t)} x_i(t) = \sum_{i\in V(t)} \bar{x}_i, \tag{3.19}$$

thus proving the theorem. ∎

To carry out the convergence analysis for Algorithm 1 we now study the so-called error dynamics of the agents with respect to the desired convergence value.

Proposition 3.5.1

Consider a MAS which executes Algorithm 1. Let us call the error $e_i(t) = x_i(t) - c(t)$ as disagreement value. According to Algorithm 1 when an agent joins the network at time t it holds $x_i(t) = \bar{x}_i$. Then, the error dynamics of remaining and arriving agents can obtain as follows:

$$\begin{aligned}
e_i(t+1) =& e_i(t) + \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big) - \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\\
&+ c(t) - c(t+1), \quad i \in R(t).
\end{aligned} \tag{3.20}$$

$$e_i(t+1) = \bar{x}_i(t) - c(t+1), \quad i \in A(t). \tag{3.21}$$

Proof of Proposition 3.5.1 *The disagreement error for agent i is:*

$$e_i(t) = x_i(t) - c(t) \tag{3.22}$$

Thus, the time-shift of the disagreement error yields as:

$$e_i(t+1) = \quad x_i(t+1) - c(t+1). \tag{3.23}$$

Now, by substituting Eq. (3.5) *into* (3.23)*, we can obtain:*

$$e_i(t+1) = x_i(t) + \sum_{j \in N_i(t) \setminus D(t)} w_{ij}(t)\big(x_j(t) - x_i(t)\big) - \sum_{j \in N_i(t) \cap D(t)} z_{ij}(t) - c(t+1) \tag{3.24}$$

By adding to the right hand side of Eq. (3.24) *the term* $\pm c(t)$*, one can get the updating error dynamics of remaining agents, i.e.,* $i \in R(t)$ *as follow:*

$$\begin{aligned} e_i(t+1) =& x_i(t) - c(t) + \sum_{j \in N_i(t) \setminus D(t)} w_{ij}(t)\big(x_j(t) - x_i(t)\big) \\ &+ c(t) - c(t+1) - \sum_{j \in N_i(t) \cap D(t)} z_{ij}(t), \end{aligned} \tag{3.25}$$

Then, by noting that $x_j(t) - x_i(t) = x_j(t) - c(t) - x_i(t) + c(t) = e_j(t) - e_i(t)$*, it holds:*

$$\begin{aligned} e_i(t+1) =& e_i(t) + \sum_{j \in N_i(t) \setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big) \\ &+ c(t) - c(t+1) - \sum_{j \in N_i(t) \cap D(t)} z_{ij}(t). \end{aligned} \tag{3.26}$$

On the other hand, according to Algorithm 1 the initial state of arriving agents, i.e., $i \in A(t)$ *is* $x_i(t) = \bar{x}_i$*. Hence, one can obtain the error dynamics of arriving agents as:*

$$x_i(t+1) - c(t+1) = \bar{x}_i - c(t+1), \tag{3.27}$$

$$e_i(t+1) = \bar{x}_i - c(t+1). \tag{3.28}$$

This concludes the proof. ■

- Part 2: The next theorem shows that the state variables of the agents executing Algorithm 1 converge exactly to the average value of the initial state of the agents if no agent joins or leaves the network. The result of the next theorem is an adaptation of a popular result of convergence for discrete-time average consensus to our scenario [62].

Theorem 3.5.2

Consider a MAS executing Algorithm 1. Let $G(t)$ be an undirected graph with a doubly stochastic matrix $P(t) = I + W(t)$. Assume that no agent arrives or departs to the network since an instant of time t. Then, the state of the agents exponentially converges to $c(t)$, i.e.

$$\|x_i(t) - c(t)\| \longrightarrow 0 \quad \forall i \quad as\ t \longrightarrow \infty \tag{3.29}$$

Proof of Theorem 3.5.2 *Let us define the disagreement value:*

$$e_i(t) = x_i(t) - c(t), \qquad i \in V(t). \tag{3.30}$$

Thus, the time-shift of the disagreement error yields as:

$$e_i(t+1) = x_i(t+1) - c(t+1). \tag{3.31}$$

According to Eq. (3.5) *and assuming no agent arrives or departs to the network (namely,* $c(t+1) = c(t)$ *and* $\sum_{j \in N_i(t) \cap D(t)} z_{ij}(t) = 0$ *, the following disagreement dynamics can be deduced:*

$$e_i(t+1) = x_i(t) + \sum_{j \in N_i(t)} w_{ij}(t)\big(x_j(t) - x_i(t)\big) - c(t), \tag{3.32}$$

Then, by noting that $x_j(t) - x_i(t) = x_j(t) - c(t) - x_i(t) + c(t) = e_j(t) - e_i(t)$*, one can get:*

$$e_i(t+1) = e_i(t) + \sum_{j \in N_i(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big) \tag{3.33}$$

Moreover, we can write the compact form of disagreement dynamics as follow:

$$e(t+1) = P(t)e(t) \tag{3.34}$$

where $e(t)$ *is a vector whose elements are taken from the set* $\{e_i(t) : i \in V(t)\}$*.*

Now consider the next Lyapunov function:

$$\mathscr{V}(t) = \sum_{i \in V(t)} e_i(t)^2, \tag{3.35}$$

so that the time-shift of $\mathscr{V}(t)$ *takes this form:*

$$\mathscr{V}(t+1) = \sum_{i \in V(t)} e_i(t+1)^2. \tag{3.36}$$

By replacing (3.33) *into* (3.36)*, it yields:*

$$\mathscr{V}(t+1) = \sum_{i \in V(t)} \Big(e_i(t) + \sum_{j \in N_i(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big)^2. \tag{3.37}$$

Then we can recast (3.37) *as follows:*

$$\begin{aligned}
\mathscr{V}(t+1) &= \sum_{i\in V(t)} \Big(e_i(t)^2 + 2e_i(t) \sum_{j\in N_i(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) \\
&\quad + \Big(\sum_{j\in N_i(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big)\Big)^2 \Big) \\
&= \sum_{i\in V(t)} e_i(t)^2 + \sum_{i\in V(t)} 2e_i(t) \sum_{j\in N_i(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) \\
&\quad + \sum_{i\in V(t)} \Big(\sum_{j\in N_i(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big)\Big)^2 \qquad (3.38)
\end{aligned}$$

Now let us compute $\Delta\mathscr{V}(t)$ *as:*

$$\begin{aligned}
\Delta\mathscr{V}(t) = \mathscr{V}(t+1) - \mathscr{V}(t) &= \sum_{i\in V(t)} 2e_i(t) \sum_{j\in N_i(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) \\
&\quad + \sum_{i\in V(t)} \Big(\sum_{j\in N_i(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big)\Big)^2 \\
&= \sum_{i\in V(t)} \sum_{j\in N_i(t)} 2e_i(t) w_{ij}(t)\big(e_j(t)-e_i(t)\big) \\
&\quad + \sum_{i\in V(t)} \Big(\sum_{j\in N_i(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big)\Big)^2. \qquad (3.39)
\end{aligned}$$

Now, we can note that the first part of (3.39) *can be expressed as:*

$$2e(t)^T W(t) e(t). \qquad (3.40)$$

We can also recast the second part of equation (3.39) *as:*

$$e(t)^T W(t)^T W(t) e(t). \qquad (3.41)$$

Hence, since $W(t) = W(t)^T$ *and from* (3.40) *and* (3.41)*, we can rewrite* (3.39) *as:*

$$\Delta\mathscr{V}(t) \quad = -e(t)^T \Big(-2W(t) - W(t)^2 \Big) e(t) \qquad (3.42)$$

Finally, by noting that:

$$\begin{aligned}
-2W(t) - W(t)^2 &= -2W(t) - W(t)^2 + I - I \\
&= I - \big(I + W(t)\big)^2 \\
&= I - P(t)^2, \qquad (3.43)
\end{aligned}$$

we can rewrite (3.39) *as:*

$$\begin{aligned}
\Delta\mathscr{V}(t) = &\quad -e(t)^T (I - P(t)^2) e(t) \\
= &\quad -(I - P(t)^2)\|e(t)\|^2.
\end{aligned} \qquad (3.44)$$

According to Corollary 2 in [62], we can write:

$$||P(t)e(t)||^2 < \mu_2^2||e(t)||^2 \tag{3.45}$$

where μ_2 is the second largest eigenvalue of matrix $P(t)$. Note that for a connected undirected network, it can be computed as $\mu_2 = 1-\lambda_2(W)$. Hence, from eq. (3.45)*, we can recast eq.* (3.44) *as:*

$$\Delta\mathscr{V}(t) \quad \leq -(1-\mu_2^2)||e(t)||^2. \tag{3.46}$$

Taking into account that $P(t)$ is a primitive and doubly stochastic matrix for all times, it derives that $0<\mu_2<1$*. Consequently, it proves $\mathscr{V}(t)$ exponentially converges to zero and thus condition* (3.29) *is in force. This concludes the proof.* ■

- Part 3. In the next theorem we prove that when agents join or leave the network, the time-varying average of the initial states can be tracked with a bounded error. Note that the changes of the consensus value is assumed to be bounded at each time step.

Theorem 3.5.3

Consider a MAS of $n(t)$ agents under the OACP presented in Algorithm 1. Let the communication topology graph $G(t)$ be characterized by a doubly stochastic matrix $P(t) = I+W(t)$ that satisfies Assumptions 3.4.1 and 3.4.2. Let $\mu_2(t)$ be the second largest eigenvalue of matrix $P(t)$. Then, let $e(t)$ be the disagreement vector with elements taken from the set $\{e_i(t) : i \in V(t)\}$. The norm of the disagreement vector is monotonically decreasing if

$$\|e(t)\|_2 \geq \Psi = \frac{(B+\|Z(t)\|_2)(1+\sqrt{1+\rho+\rho\frac{|A(t)|B^2}{(B+\|Z(t)\|_2)^2}})}{\rho}, \tag{3.47}$$

where $|c(t)-c(t+1)| \leq B,\ \rho = \max(1-\mu_2(t)^2)$ and

$$\|Z(t)\|_2^2 = \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\Big\}^2. \tag{3.48}$$

Note that $Z(t)\neq 0$ only at times t where a node departs, in all other cases $Z(t)=0$.

Proof of Theorem 3.5.3 *Let us consider the following Lyapunov function:*

$$\mathscr{V}(t) = \sum_{i\in V(t)} e_i(t)^2, \tag{3.49}$$

so that the time difference of $\mathscr{V}(t)$ correspondingly takes the form:

$$\begin{aligned}\mathscr{V}(t+1) = \sum_{i\in V(t+1)} e_i(t+1)^2 &= \sum_{i\in(V(t)\bigcup A(t))\setminus D(t)} e_i(t+1)^2 \\ &= \sum_{i\in V(t)\setminus D(t)} e_i(t+1)^2 + \sum_{i\in A(t)} e_i(t+1)^2.\end{aligned} \tag{3.50}$$

Computing $\Delta\mathscr{V}(t) = \mathscr{V}(t+1) - \mathscr{V}(t)$, yields:

$$\Delta\mathscr{V}(t) = \sum_{i\in V(t)\setminus D(t)} e_i(t+1)^2 + \sum_{i\in A(t)} e_i(t+1)^2 - \sum_{i\in V(t)} e_i(t)^2. \tag{3.51}$$

By substituting the error dynamics of remaining agents in (3.26)*, into* (3.51)*, it yields:*

$$\begin{aligned}\Delta\mathscr{V}(t) = &\sum_{i\in V(t)\setminus D(t)} \Big\{ e_i(t) + \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big) \\ &- \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) + c(t) - c(t+1)\Big\}^2 + \sum_{i\in A(t)} e_i(t+1)^2 - \sum_{i\in V(t)} e_i(t)^2.\end{aligned} \tag{3.52}$$

Then, we can write:

$$\begin{aligned}\Delta\mathscr{V}(t) = &\sum_{i\in V(t)\setminus D(t)} e_i(t)^2 + \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\}^2 \\ &+ \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\Big\}^2 + \sum_{i\in V(t)\setminus D(t)} \big(c(t) - c(t+1)\big)^2 \\ &+ 2\sum_{i\in V(t)\setminus D(t)} \Big\{ e_i(t) \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\} \\ &- 2\sum_{i\in V(t)\setminus D(t)} \Big\{ e_i(t) \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\Big\} + 2\sum_{i\in V(t)\setminus D(t)} \Big\{ e_i(t)\big(c(t) - c(t+1)\big)\Big\} \\ &- 2\sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\} \\ &+ 2\sum_{i\in V(t)\setminus D(t)} \Big\{ \big(c(t) - c(t+1)\big) \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\} \\ &- 2\sum_{i\in V(t)\setminus D(t)} \Big\{ \big(c(t) - c(t+1)\big) \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\Big\} \\ &+ \sum_{i\in A(t)} e_i(t+1)^2 - \sum_{i\in V(t)} e_i(t)^2.\end{aligned} \tag{3.53}$$

By adding $\pm \sum_{i \in D(t)} e_i(t)^2$ *to* (3.53)*, one can concludes:*

$$\begin{aligned}
\Delta\mathscr{V}(t) = &- \sum_{i \in D(t)} e_i(t)^2 + \sum_{i \in A(t)} e_i(t+1)^2 + \sum_{i \in V(t)\setminus D(t)} \Big\{ \sum_{j \in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\}^2 \\
&+ \sum_{i \in V(t)\setminus D(t)} \Big\{ \sum_{j \in N_i(t)\cap D(t)} z_{ij}(t)\Big\}^2 + \sum_{i \in V(t)\setminus D(t)} \big(c(t) - c(t+1)\big)^2 \\
&+ 2 \sum_{i \in V(t)\setminus D(t)} \Big\{ e_i(t) \sum_{j \in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\} \\
&- 2 \sum_{i \in V(t)\setminus D(t)} \Big\{ e_i(t) \sum_{j \in N_i(t)\cap D(t)} z_{ij}(t)\Big\} + 2 \sum_{i \in V(t)\setminus D(t)} \Big\{ e_i(t)\big(c(t) - c(t+1)\big)\Big\} \\
&- 2 \sum_{i \in V(t)\setminus D(t)} \Big\{ \sum_{j \in N_i(t)\cap D(t)} z_{ij}(t) \sum_{j \in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\} \\
&+ 2 \sum_{i \in V(t)\setminus D(t)} \Big\{ \big(c(t) - c(t+1)\big) \sum_{j \in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\} \\
&- 2 \sum_{i \in V(t)\setminus D(t)} \Big\{ \big(c(t) - c(t+1)\big) \sum_{j \in N_i(t)\cap D(t)} z_{ij}(t)\Big\}. \qquad (3.54)
\end{aligned}$$

It can be noticed that the term

$$\begin{aligned}
&\sum_{i \in V(t)\setminus D(t)} \Big\{ \sum_{j \in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\}^2 \\
&+ 2 \sum_{i \in V(t)\setminus D(t)} \Big\{ e_i(t) \sum_{j \in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t) - e_i(t)\big)\Big\}.
\end{aligned}$$

have the same structure as (3.44)*. Let* $P(t)$ *denote the weight matrix of the network at time* t*. Let* $P_D(t)$ *denote a matrix representing the weights of the network graph at time* t *where the edges corresponding to departing nodes and the departing nodes have been removed. It follows that the essential spectral radius of* $P_D(t)$ *is strictly less than the essential spectral radius of matrix* $P(t)$ *which is strictly less than* 1*. Thus, it holds:*

$$-e(t)^T (I - P_D(t)^2) e(t) \le -\rho e(t)^T e(t), \qquad (3.55)$$

with $\rho = \max(1 - \mu_2(t)^2)$*. Where* $\mu_2(t)$ *is the second largest eigenvalue of matrix* $P_D(t)$ *and is different for every graph. However, since* $P_D(t)$ *is a primitive matrix for all times and there is a maximum number of nodes according to Assumption 3.4.2, it derives that* $0 < \mu_2(t) < 1$*. Consequently, it follows that* $0 < \rho < 1$ *and Eq.* (3.54) *can be rewritten* *as:*

$$\begin{aligned}\Delta\mathscr{V}(t) \leq &-\sum_{i\in D(t)} e_i(t)^2 - \rho \sum_{i\in V(t)\setminus D(t)} e_i^2(t)2 + \sum_{i\in A(t)} e_i(t+1)^2 \\ &+ \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\}^2 + \sum_{i\in V(t)\setminus D(t)} \big(c(t)-c(t+1)\big)^2 \\ &-2 \sum_{i\in V(t)\setminus D(t)} \Big\{ e_i(t) \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\} + 2 \sum_{i\in V(t)\setminus D(t)} \Big\{ e_i(t)\big(c(t)-c(t+1)\big) \Big\} \\ &-2 \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) \Big\} \\ &+2 \sum_{i\in V(t)\setminus D(t)} \Big\{ \big(c(t)-c(t+1)\big) \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) \Big\} \\ &-2 \sum_{i\in V(t)\setminus D(t)} \Big\{ \big(c(t)-c(t+1)\big) \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\}. \end{aligned} \tag{3.56}$$

By considering $|c(t)-c(t+1)| \leq B$*, we can rewrite* (3.56) *as:*

$$\begin{aligned}\Delta\mathscr{V}(t) \leq &-\sum_{i\in D(t)} e_i(t)^2 - \rho \sum_{i\in V(t)\setminus D(t)} e_i^2(t) + \sum_{i\in A(t)} e_i(t+1)^2 + \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\}^2 \\ &+ B^2 - 2 \sum_{i\in V(t)\setminus D(t)} \Big\{ e_i(t) \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\} + 2B \sum_{i\in V(t)\setminus D(t)} e_i(t) \\ &-2 \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) \Big\} \\ &+2B \sum_{i\in V(t)\setminus D(t)} \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) - 2B \sum_{i\in V(t)\setminus D(t)} \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t). \end{aligned} \tag{3.57}$$

Exploiting the symmetry of interactions since we have an undirected graph, it holds:

$$\sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) \Big\} = 0. \tag{3.58}$$

Thus, we can write (3.57) *as:*

$$\begin{aligned}\Delta\mathscr{V}(t) \leq & -\sum_{i\in D(t)} e_i(t)^2 - \rho \sum_{i\in V(t)\setminus D(t)} e_i^2(t) + \sum_{i\in A(t)} e_i(t+1)^2 + \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\}^2 \\ & + B^2 - 2 \sum_{i\in V(t)\setminus D(t)} \Big\{ e_i(t) \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\} + 2B \sum_{i\in V(t)\setminus D(t)} e_i(t) \\ & - 2 \sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big) \Big\} \\ & - 2B \sum_{i\in V(t)\setminus D(t)} \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \end{aligned} \tag{3.59}$$

Now let us define $Z(t) \in \mathbb{R}^{|R(t)|}$ *as a vector whose elements are taken from the set* $\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) : i \in V(t) \setminus D(t)\}$*. Hence, one concludes that:*

$$\sum_{i\in V(t)\setminus D(t)} \Big\{ \sum_{j\in N_i(t)\cap D(t)} z_{ij}(t) \Big\}^2 = ||Z(t)||_2^2 \tag{3.60a}$$

By using the following property:

$$\sum_{i=i_0}^{q} a^2 \leq \Big(\sum_{i=i_0}^{q} |a| \Big)^2 \tag{3.60b}$$

one gets:

$$\|Z(t)\|_2^2 \leq \Big\{ \sum_{i\in V(t)\setminus D(t)} \sum_{j\in N_i(t)\cap D(t)} |z_{ij}(t)| \Big\}^2 \tag{3.60c}$$

From Eq. (3.14)*, we can recast* (3.60c) *as:*

$$\|Z(t)\|_2^2 \leq \Big(\sum_{(i,j)\in E_D(t)} |z_{ij}(t)| \Big)^2 \tag{3.60d}$$

Then, according to (3.60d) *and* (3.16) *the upper bound of* $\|Z(t)\|_2$ *can be yield as:*

$$\|Z(t)\|_2 \leq \sum_{i\in D(t)} |x_i(t)| \tag{3.60e}$$

It should be noted that when the state of agents converge to the consensus value, namely $x_i(t) = x_j(t)$*, the term* $x_i(t) - x_j(t)$ *goes to zero, then it results that the auxiliary state variable* $z_{ji}(t)$ *converges to the finite value, see eq.* (3.6)*. Consequently, it yields* $\|Z(t)\|_2$ *be bounded.*

Moreover, we can write:

$$
\begin{aligned}
&-2\sum_{i\in V(t)\setminus D(t)}\Big\{e_i(t)\sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\Big\}\leq \\
&2\Big\{\Big(\sum_{i\in V(t)\setminus D(t)} e_i(t)^2\Big)^{\frac{1}{2}}\times\Big(\sum_{i\in V(t)\setminus D(t)}\Big(\sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\Big)^2\Big)^{\frac{1}{2}}\Big\} \\
&=2\Big(\sum_{i\in V(t)\setminus D(t)} e_i(t)^2\Big)^{\frac{1}{2}}||Z||_2
\end{aligned} \tag{3.61}
$$

we also write:

$$
\begin{aligned}
&-2\sum_{i\in V(t)\setminus D(t)}\Big\{\sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\times\sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big)\Big\}\leq \\
&2\Big\{\Big(\sum_{i\in V(t)\setminus D(t)}\Big(\sum_{j\in N_i(t)\cap D(t)} z_{ij}(t)\Big)^2\Big)^{\frac{1}{2}}\times\Big|\sum_{i\in V(t)\setminus D(t)}\sum_{j\in N_i(t)\setminus D(t)} w_{ij}(t)\big(e_j(t)-e_i(t)\big)\Big|\Big\}=0
\end{aligned} \tag{3.62}
$$

Hence, according equations (3.58) *and* (3.60)*, we can rewrite* (3.59) *as:*

$$
\begin{aligned}
\Delta\mathscr{V}(t)\leq&-\sum_{i\in D(t)} e_i(t)^2-\rho\sum_{i\in V(t)\setminus D(t)} e_i^2(t)+\sum_{i\in A(t)} e_i(t+1)^2+||Z||_2^2+B^2 \\
&+2\Big(\sum_{i\in V(t)\setminus D(t)} e_i(t)^2\Big)^{\frac{1}{2}}||Z||_2+2B\Big(\sum_{i\in V(t)\setminus D(t)} e_i(t)^2\Big)^{\frac{1}{2}}+2B||Z||_2.
\end{aligned} \tag{3.63}
$$

Then, by noting that $-\rho\sum_{i\in D(t)} e_i(t)^2\geq-\sum_{i\in D(t)} e_i(t)^2$ *and* $\sum_{i\in V(t)\setminus D(t)} e_i(t)^2\leq\sum_{i\in V(t)} e_i(t)^2$*, Eq.* (3.63) *can be recast as:*

$$
\begin{aligned}
\Delta\mathscr{V}(t)\leq&-\rho\sum_{i\in V(t)} e_i^2(t)+\sum_{i\in A(t)} e_i(t+1)^2\quad+||Z||_2^2+B^2+2\Big(\sum_{i\in V(t)} e_i(t)^2\Big)^{\frac{1}{2}}||Z||_2 \\
&+2B\Big(\sum_{i\in V(t)} e_i(t)^2\Big)^{\frac{1}{2}}+2B||Z||_2.
\end{aligned} \tag{3.64}
$$

Now, If we let $\sum_{i\in V(t)} e_i^2(t)=\|e(t)\|_2^2=\Pi^2$ *and* $||Z||_2=\Gamma$*, we find that:*

$$\Delta\mathscr{V}(t)=-\rho\Pi^2+2(B+\Gamma)\Pi+(B+\Gamma)^2+\sum_{i\in A(t)} e_i(t+1)^2. \tag{3.65}$$

By substituting (3.21) *into* (3.65)*, one can obtain that:*

$$\Delta\mathscr{V}(t)\leq-\rho\Pi^2+2(B+\Gamma)\Pi+(B+\Gamma)^2+\sum_{i\in A(t)}(\bar{x}_i-c(t+1))^2. \tag{3.66}$$

Then, assuming $\{|\bar{x}_i - c(t+1)| \leq B : i \in A(t)\}$*, finally* $\Delta\mathscr{V}(t)$ *can be concluded as:*

$$\Delta\mathscr{V}(t) \leq -\rho\Pi^2 + 2(B+\Gamma)\Pi + (B+\Gamma)^2 + |A(t)|B^2. \tag{3.67}$$

Since ρ*,* B*,* Γ *and* $|A(t)|$ *are strictly positive, we have:*

$$-\rho\Pi^2 + 2(B+\Gamma)\Pi + (B+\Gamma)^2 + |A(t)|B^2 = 0 \tag{3.68}$$

Then, we can obtain the solution of (3.68) *as:*

$$\Pi = \frac{-2(B+\Gamma) \pm \sqrt{4(B+\Gamma)^2 + 4\rho\left((B+\Gamma)^2 + |A(t)|B^2\right)}}{-2\rho} = \frac{(B+\Gamma)(1 \pm \sqrt{1+\rho+\rho\frac{|A(t)|B^2}{(B+\Gamma)^2}})}{\rho}. \tag{3.69}$$

Thus, if $\|e(t)\|_2 = \Pi \geq \frac{(B+\Gamma)(1+\sqrt{1+\rho+\rho\frac{|A(t)|B^2}{(B+\Gamma)^2}})}{\rho}$*, then* $\Delta\mathscr{V}(t) < 0$*. This concludes the proof.* ■

Corollary 3.5.1

Now, we show that when the set of agents changes, the value of the considered Lyapunov function (3.49) how much can be changed.
Let at time $\hat{t}$ the leave event only occurs. From (3.69) and taking account into $A(\hat{t})$ is empty, we can obtain

$$\|e(\hat{t})\| = \frac{(B+\Gamma)(1+\sqrt{1+\rho})}{\rho} \tag{3.70}$$

Thus, according to (3.49) we can write

$$\mathscr{V}(\hat{t}) \leq \frac{(B+\Gamma)^2(1+\sqrt{1+\rho})^2}{\rho^2} \tag{3.71}$$

If, instead, agents join the network, in accordance with (3.69) the following relation can be yield:

$$\|e(\hat{t})\| = \frac{B(1+\sqrt{1+\rho+\rho\frac{|A(t)|B^2}{B^2}})}{\rho} \tag{3.72}$$

Hence, it results that:

$$\mathscr{V}(\hat{t}) \leq \frac{B^2\left(1+\sqrt{1+\rho+\rho\frac{|A(t)|B^2}{B^2}}\right)^2}{\rho^2} \tag{3.73}$$

Remark 3.5.1

It should be remarked that by Theorem 3.5.3 the consensus value of the MAS has no bias, and due to the result in Theorem 3.5.3 the disagreement error is bounded, thus if agents stop joining or leaving the network for a sufficiently long time, the state of the network converges exactly to the average of the initial states of the agents, erasing the influence on the consensus value of the agents who left the network.

3.6 Numerical simulations

In this section, two numerical examples are conducted to evaluate the efficiency of the proposed algorithm. In the first scenario, we consider a multi-agent system consisting of 6 agents that interact over a network by a random graph. The initial information states $x_i(0)$ are selected uniformly at random in the interval $[0,6]$. Simulations were performed in the MATLAB® environment. The time evolution of the state variables are depicted in Figure 3.1. It can be seen from this figure that the state of agents converge to the average of initial information states.

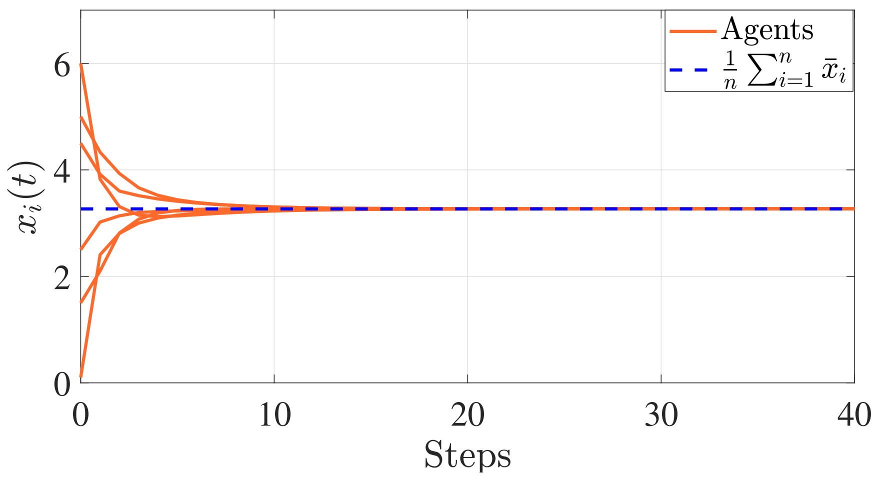

Figure 3.1: Time evolution of the network state $x_i(t)$ for all $i \in V(t)$.

In the second example, we consider an open multi-agent system where simulation starts with the same communication topology of the first scenario, then at $t = 100$ sec an agent leaves the network. Moreover, at $t = 200$ sec an agent joins the network with initial state equal to 4. It is obvious from Figure 3.2 when an agent leaves the network, the trajectories of all agents that interact on the network, track successfully the average value of their information at the instant of joining.

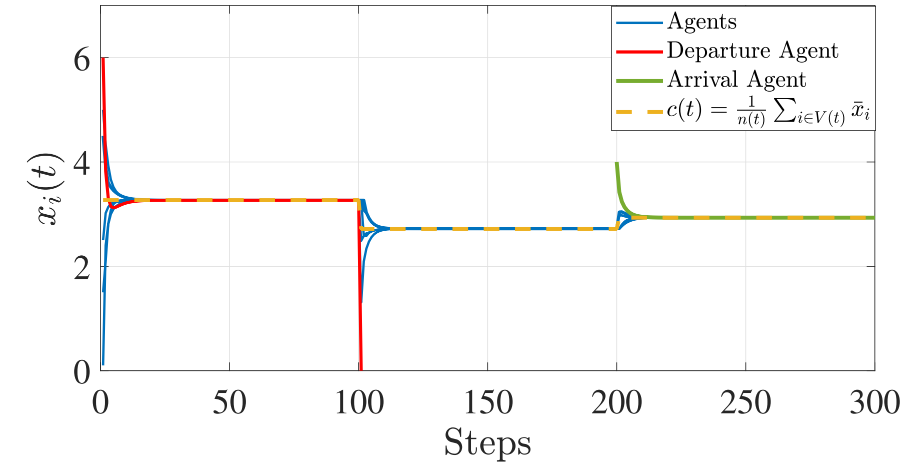

Figure 3.2: Time evolution of the network state $x_i(t)$ for all $i \in V(t)$ during the join and leave events.

Furthermore, as expected, when an agent arrives in the network, the state of agents remains at the desired consensus value. Figure 3.3 depicts time evolution of the sum of the state of agents during the join and leave events. It can be seen from this figure, the condition (3.8) is always established. The time evolution of Lyapunouv function

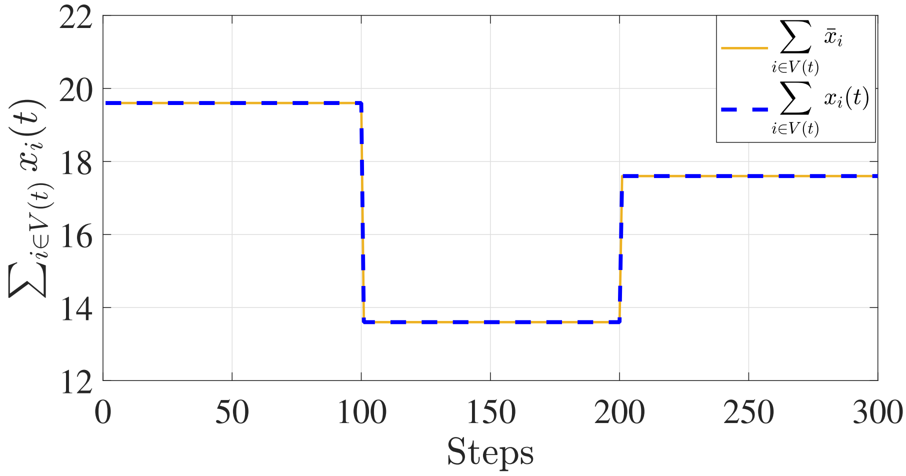

Figure 3.3: The summation of the state of agents $\sum_{i \in V(t)} x_i(t)$ during the join and leave events.

$\mathscr{V}(t)$ is shown in Figure 3.4. It is clear from Figures 3.4-3.6 that $\mathscr{V}(t)$ is monotonically decreasing as long as $\|e(t)\| \geq \Psi(t)$, whereas if the norm of $e(t)$ is less than $\Psi(t)$, the Lyapunov function may increase or decrease. As a conclusion, the proposed algorithm offers robustness against the join and leave events and the system always achieves average consensus. However, the agreement value is depending on the information states at the instant of adding the network.

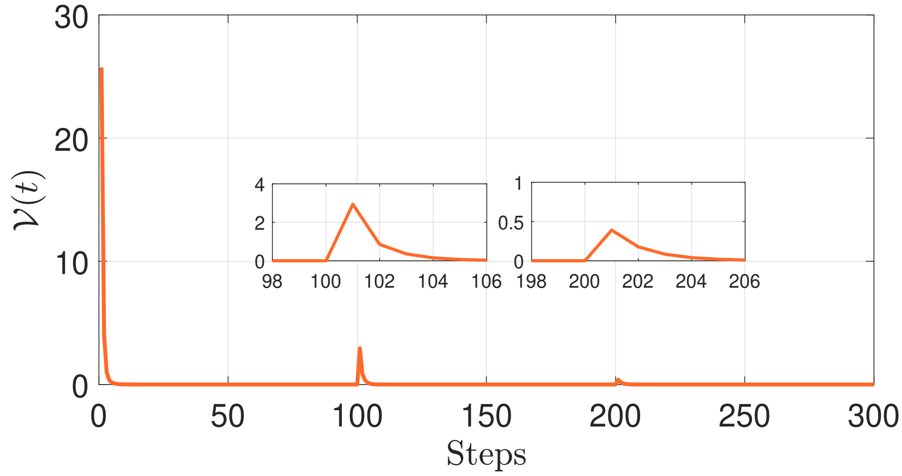

Figure 3.4: Time evolution of the Lyapunov function $\mathscr{V}(t) = \sum_{i \in V(t)} e_i(t)^2$.

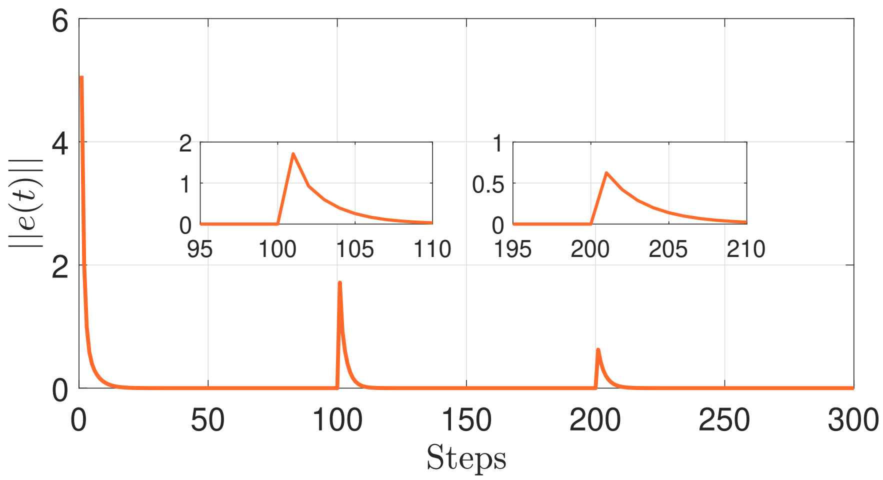

Figure 3.5: Time evolution of the norm of error $||e(t)||$.

3.7 Conclusions

In this chapter, we considered the problem of average discrete time consensus in an open multi-agent system under a switching network topology. This problem solved by designing a novel distributed open average consensus protocol that guarantees that the state variables of the agents track the average of their initial state despite a network with

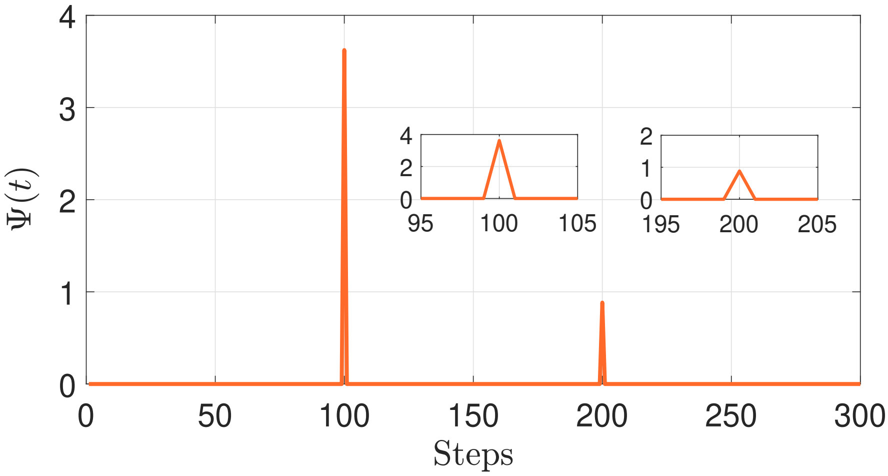

Figure 3.6: Time evolution of the threshold value $\Psi(t)$.

time-varying size and composition. Future works will focus on the addition of resilience with respect to faults or malicious agents to the proposed method.

Chapter 4

Dynamic Consensus on the Median Value

4.1 Introduction

This chapter deals with the problem of dynamic consensus of a network of agents corresponding to continuous-time systems wherein the state variables of the agents track with zero error the median value of a set of time-varying reference signals given as input to the agents under a time-varying, undirected network topology. Furthermore, the performance of the protocol in the framework of open multi-agent systems is analysed by considering join and leave events, i.e., the scenario where agents may join and leave the network during the protocol execution. The finite-time convergence properties and tracking error of the considered protocol are characterized in the case of inputs with bounded variations. One remarkable property of consensus on the median value is the robustness of the median, as opposed to the average, with respect to abnormal or outlier values of inputs which represent the outcome of a measurement or estimation process, thus significantly increasing the robustness of the estimation for large scale networks. We provide convergence guarantees and simple tuning rules to adjust the algorithm parameters by employing non-smooth Lyapunov analysis.

The chapter is organized as follows: In Section 4.2 we present the contributions of this chapter. The problem statement is given in Section 4.3. In this section we also introduce the dynamic consensus algorithm. In Section 4.4 we characterize the convergence properties of the algorithm in both dynamic and open scenarios by exploiting non-smooth Lyapunov tools. In Section Section 4.5 we provide numerical simulations. Finally in Section 4.6 concluding remarks and future perspectives are discussed.

4.2 Main contributions

The main contributions of this work are as follows: i) We characterize the performance of a variant of the local interaction rule in [55] to solve a dynamic consensus problem on the median value. ii) We consider the case in which the network topology is time-varying and connected at all times as opposed to time-invariant. iii) We show that if the reference signals of the agents have a bounded derivative, even if the median value is not unique due to their even number, then the average of median values is continuous and is tracked in finite time by the agents. iv) We consider an open-multi-agent scenario where agents join and leave the network and characterize the convergence properties of the method. v) We show that in the open scenario the time-varying median value is discontinuous, and the approach is able to track it with bounded error.

4.3 Problem statement and dynamic consensus algorithm

In this subsection, we consider a continuous-time open MAS with the network topology represented by an undirected graph $G(t) = (V(t), E(t))$ where the agents have dynamics described by:

$$\dot{x}_i(t) = -\lambda \sum_{j \in \mathcal{N}_i(t)} \text{sign}\left(x_i(t) - x_j(t)\right) - \alpha \, \text{sign}\left(x_i(t) - u_i(t)\right), \tag{4.1}$$

where $x_i(t) \in \mathbb{R}$ is the state of the i-th agent and $u_i(t)$ is a local reference signal given as input, which represents a measurement or, more generally, an exogenous time-varying information. λ and α are the tunable protocol constant gains. Moreover, the sign function in (4.1) for $y \in \mathbb{R}$, is denoted as:

$$\text{sign}(y) = \begin{cases} 1, \text{ if } y > 0, \\ 0, \text{ if } y = 0, \\ -1, \text{ if } y < 1. \end{cases} \tag{4.2}$$

When an agent joins the network at time t it holds:

$$x_i(t) = \frac{1}{|\mathcal{N}_i(t)|} \sum_{j \in \mathcal{N}_i(t)} x_j(t). \tag{4.3}$$

When an agent leaves the network, its state variable and incident edges are removed from the system.

In the following, we formalize the definition of median value.

Definition 4.3.1: Median value

The median value of a set of real numbers is a number which is greater than half of the numbers in the set and smaller than the other half. The median value is not uniquely defined when the cradinality of the set is even. Let us compactly define $u(t) = [u_1(t), \ldots, u_n(t)]$, as the vector set of reference signals at time t and then let

$$\chi(t) = \arg\min_{\bar{u}(t)\in\mathscr{U}(t)} \sum_{j\in\mathscr{V}(t)} |u_j(t) - \bar{u}(t)|, \tag{4.4}$$

where $\mathscr{U}(t)$ is the set of all entries of vector $u(t)$. If the number of agents at time t is odd, then the solution of (4.4), i.e., the median value, is unique and set $\chi(t)$ is a singleton. On the other hand, if at time t the number of agents is even, then $\chi(t)$ contains two solutions, both with minimum objective value. By averaging the elements of set $\chi(t)$ we define the (unique) m-value $m(u(t))$ of vector $u(t)$ as

$$m(u(t)) = \frac{1}{|\chi(t)|} \sum_{\xi_r\in\chi(t)} \xi_r, \tag{4.5}$$

where $|\chi(t)|$ denotes the cardinality of the set. We also define the so-called median interval $\mathscr{M}(u(t))$ as

$$\mathscr{M}(u(t)) = \left[\min_{\xi_r\in\chi(t)} \xi_r, \max_{\xi_r\in\chi(t)} \xi_r\right]. \tag{4.6}$$

Any number inside the median interval $\mathscr{M}(u(t))$ is a median value.
We also define by $L_M(t)$ the time-varying length of the median interval

$$L_M(t) = \max_{\xi_r\in\chi(t)} \xi_r - \min_{\xi_r\in\chi(t)} \xi_r. \tag{4.7}$$

Clearly $L_M(t) = 0$ if the number of agents is odd at time t.

To clarify the above definitions, consider the following example.

Example 4.3.1

In the first example, let us consider the number of agents, n, is odd and equal to 5 with

$$\mathscr{U}(t) = \{u_1 = 3, u_2 = 5, u_3 = 6, u_4 = 7.5, u_5 = 8\}.$$

According to (4.4), the solution of $\chi(t)$ is a singleton, and is equal to $\chi(t) = \{u_3\} = \{6\}$. Hence, it results that $m(u(t)) = 6$. Moreover, from (4.6) and (4.7) the median interval and the length of the median value are obtained 6 and 0, respectively.
Now let us assume the number of agents is odd, and is equal to 6 with

$$\mathscr{U}(t) = \{u_1 = 3, u_2 = 5, u_3 = 6, u_4 = 7, u_5 = 8, u_6 = 9\}.$$

In accordance with the definition of $\chi(t)$, we can find two solutions as $\chi(t) = \{u_3 = 6, u_4 = 7\}$. Thus, median value is $m(u(t)) = 6.5$.In addition, Median interval and Length of the median interval are computed as $\mathscr{M}(u(t)) = \left[\min_{\xi_r \in \chi(t)} \xi_r, \max_{\xi_r \in \chi(t)} \xi_r\right] = [6, 7]$ and $L_M(t) = \max_{\xi_r \in \chi(t)} \xi_r - \min_{\xi_r \in \chi(t)} \xi_r = 7 - 6 = 1$.

In this chapter, our objective is to verify that the proposed local interaction in (4.1) enables the state of agents to track the time-varying median value of reference signals $m(u_i(t))$ in finite-time under both the frameworks of time-invariant network and time-varying network, i.e., the case where agents may join and leave the network as time goes by. To this aim, we will firstly indicate that the state of agents reaches $c(t)$ in finite time in both close and open scenarios, i.e.,

$$\|x_i(t) - c(t)\| \longrightarrow 0 \quad \forall i \in V(t), \quad \forall t \geq T. \tag{4.8}$$

Then, we will show if the composition of network is time-invariant (time-varying), the time-varying median value of reference signals $m(u_i(t))$ can be tracked with zero error (with bounded error) by the state of agents in finite time, namely,

$$\|c(t) - m(u_i(t))\| \longrightarrow 0 \quad \forall i \in V(t), \quad \forall t \geq T_2. \tag{4.9}$$

In this chapter, we make the following assumption which is common in the dynamic consensus literature.

Assumption 4.3.1

All reference signals have bounded derivative, namely

$$\exists \quad \Pi \in \mathbb{R}^+ : \qquad |\dot{u}_i(t)| \leq \Pi \qquad \forall i \in \mathscr{V}(t). \tag{4.10}$$

The above assumption directly leads to the next result.

Proposition 4.3.1

If Assumption 3.2.1 holds, then

$$|\dot{m}(u(t))| \leq \Pi. \tag{4.11}$$

Proof of Proposition 4.3.1 *Let* $\Upsilon(t) = \{i \in \mathscr{V}(t) : u_i(t) = \xi_r,\ \xi_r \in \chi(t)\}$. *By eq.* (4.5) *it follows that* $\dot{m}(u(t)) = \frac{1}{|\Upsilon(t)|} \sum_{i \in \Upsilon(t)} \dot{u}_i(t)$*, and by Assumption 4.3.1*

$$|\dot{m}(u(t))| = \left| \frac{1}{|\Upsilon(t)|} \sum_{i \in \Upsilon(t)} \dot{u}_i(t) \right| \leq \frac{1}{|\Upsilon(t)|} \sum_{i \in \Upsilon(t)} \Pi = \Pi. \tag{4.12}$$

■

We also assume that the input signals satisfy the following assumption.

Assumption 4.3.2

There exists $B \in \mathbb{R}^+$ such that, for all $t \geq 0$, it holds

$$\max_{i \in \mathscr{V}(t)} (u_i(t)) - \min_{i \in \mathscr{V}(t)} (u_i(t)) \leq B. \tag{4.13}$$

Furthermore, we consider the following three assumptions on the network topology and on the way it may change.

Assumption 4.3.3

Graph $G(t)$ is undirected and connected for all $t \geq 0$ and the set of instants of time in which $G(t)$ changes is of measure zero.

Assumption 4.3.4

There exists a minimium dwell time $\Delta\tau$ separating two consecutive instants in which agents join or leave the network.

Assumption 4.3.5

There exists $n_{max} \in (0, \infty) : |\mathscr{V}(t)| \leq n_{max}$, $\forall t \geq 0$.

4.4 Finite-time convergence properties

In this section, the convergence properties of the states of a set of agents following protocol (4.1) are characterized. In particular we want to indicate that using the proposed protocol (4.1), the consensus conditions (4.8) and (4.9) can met at a finite time for both fixed and time-varying networks. To this aim, we first demonstrate in Theorem 4.4.1, the consensus condition (4.8) is verified in a finite-time T_1. In particular, T_1 is a function of the tuning parameters α and λ of the proposed local interaction rule (4.1). Then, we prove in Theorem 4.4.2 that the consensus condition (4.9) can be satisfied with zero error in a finite-time $T_2 > T_1$ despite a switching network topology if no agent departs or arrives the network. Moreover, Theorem 4.4.3 proves that condition (4.9) can be obtained with bounded error in a finite-time T_3 in the framework of open scenario.

The first main result of this chapter is now outlined.

Theorem 4.4.1

Consider the network dynamics (4.1) along with a time-varying graph $G(t)$ satifying Assumption 4.3.3. Agents may join or leave the network according to

Assumptions 4.3.4 and 4.3.5. Let the tuning parameters satisfy $0 < \alpha < \frac{2\lambda}{n_{max}}$. Then, the consensus condition (4.8) is achieved in a finite time T upper bounded as:

$$T \leq T_1 = \frac{\max_{i\in\mathscr{V}(t)} x_i(0) - \min_{i\in\mathscr{V}(t)} x_i(0)}{\mu^2}, \ \ \mu^2 = 2\left(\frac{2\lambda}{n_{max}} - \alpha\right). \tag{4.14}$$

Proof of Theorem 4.4.1 *[55] has been already mentioned a non-smooth Lyapunov function to prove finite-time convergence for a version of protocol* (4.1) *where inputs are constants, the network topology is static and agents may not leave or join the network. Here, we extend the proof in [55] to account for a switching network topology, dynamic inputs and a scenario of open multi-agent systems. To this aim, let us consider the following non-smooth Lyapunov function*

$$\mathscr{V}_1(x(t)) = \frac{\sum_{i\in I_{max}(t)} x_i(t)}{|I_{max}(t)|} - \frac{\sum_{i\in I_{min}(t)} x_i(t)}{|I_{min}(t)|}, \tag{4.15}$$

where the sets of $I_{max}(t)$ and $I_{max}(t)$ are defined as follows:

$$I_{max}(t) = \{k \in V(t) : x_k = \max_{i\in\mathscr{V}(t)} x_i(t)\},$$
$$I_{min}(t) = \{k \in V(t) : x_k = \min_{i\in\mathscr{V}(t)} x_i(t)\}.$$

By Assumptions 4.3.3 and 4.3.4, the instants of time at which graph $G(t)$ changes define a set of measure zero, i.e., there can be an infinite but countable number of such instants of time. Therefore, since the derivatives of the state variables in (4.1) *are bounded for almost all t, it follows that the state trajectories of* (4.1) *are absolutely continuous. Thus, also the sets $I_{max}(t)$ and $I_{min}(t)$ may change cardinality only at isolated instants of time. It follows that these time instants can be disregarded in the non-smooth Lyapunov analysis while evaluating the generalized gradient of $\mathscr{V}_1(x(t))$.*

First, let us present the obtained set-valued generalized time-derivative $\frac{d}{dt}(\mathscr{V}_1(x(t))$ in [55] using the definition of Clarke's generalized gradient on the non-smooth Lyapunov function, wherein $\mathscr{G}(t)$ is fixed and the set of agents does not change.

Following [55], owing on the local Lipschitz continuity of $\mathscr{V}_1(x(t))$, we can write

$$\frac{d}{dt}(\mathscr{V}_1(x(t)) = \lim_{h\to 0} \frac{\mathscr{V}_1(x(t) + h\dot{x}(t)) - \mathscr{V}_1(x(t))}{h}. \tag{4.16}$$

By substituting (4.15) *into* (4.16)*, it holds:*

$$\frac{d}{dt}(\mathscr{V}_1(x(t)) = \lim_{h\to 0}\left\{\frac{\sum_{i\in I_{max}} x_i(t) + h\dot{x}_i(t)}{h|I_{max}|} - \frac{\sum_{i\in I_{min}} x_i + h\dot{x}_i(t)}{h|I_{min}|} - \frac{\sum_{i\in I_{max}} x_i(t)}{h|I_{max}|} + \frac{\sum_{i\in I_{min}} x_i(t)}{h|I_{min}|}\right\}, \tag{4.17}$$

Then by referencing the discontinuous collective dynamics (4.1) *and due to the symmetry of interactions,* (4.17) *can be yielded as:*

$$\begin{aligned}\frac{d}{dt}\left(V_1(x(t))\right) \in\ & \frac{1}{|I_{max}|}\sum_{i\in I_{max}}\Big(-\alpha\ \mathrm{SIGN}\left(x_i(t)-u_i(t)\right)-\sum_{j\in \mathrm{N}_i(t)\setminus I_{max}}\lambda \mathrm{SIGN}\left(x_i(t)-x_j(t)\right)\Big)\\ & -\frac{1}{|I_{min}|}\sum_{i\in I_{min}}\Big(-\alpha\ \mathrm{SIGN}\left(x_i(t)-u_i(t)\right)-\sum_{j\in \mathrm{N}_i(t)\setminus I_{min}}\lambda \mathrm{SIGN}\left(x_i(t)-x_j(t)\right)\Big)\end{aligned} \tag{4.18}$$

In accordance with the definition of set I_{max}, we note that there exists every agent j in the set $\mathrm{N}_i\setminus I_{max}$ such that if the network does not reach at consensus then $x_i(t) > x_j(t)$ $\forall i\in I_{max}$, which implies that $\mathrm{SIGN}\left(x_i(t)-x_j(t)\right)=1\ \forall i\in I_{max}$. *Likewise, one concludes that* $\mathrm{SIGN}\left(x_i(t)-x_j(t)\right)=-1\ \forall i\in I_{min}$, *where the set-valued function* SIGN *for $y\in\mathbb{R}$ is defined as follow:*

$$\mathrm{SIGN}(y)\in\begin{cases}1, \text{ if } y>0,\\ \{-1,1\}, \text{ if } y=0,\\ -1, \text{ if } y<0.\end{cases} \tag{4.19}$$

Therefore, the following estimate can be derived by (4.18)

$$\frac{d}{dt}\left(\mathscr{V}_1(x(t))\right)\leq\ 2\alpha-\frac{\lambda}{|I_{max}|}-\frac{\lambda}{|I_{min}|}. \tag{4.20}$$

Moreover, if the network is not in the consensus state then $|I_{max}|=p<n$ and $|I_{min}|\leq n-p$. Hence, (4.20) *can be recast as follows:*

$$\frac{d}{dt}\left(\mathscr{V}_1(x(t))\right)\leq 2\alpha-\frac{n\lambda}{p(n-p)}. \tag{4.21}$$

By taking $p=\frac{n}{2}$, the upper bound is reached and finally, (4.21) *can be rewritten as:*

$$\frac{d}{dt}\left(\mathscr{V}_1(x(t))\right)\leq -\mu^2 \tag{4.22}$$

where μ^2 is the strictly positive constant computed in [55] and equal to $\mu^2=2\left(\frac{2\lambda}{n}-\alpha\right)$. The above estimate straightforwardly yields the finite-time convergence of $\mathscr{V}_1(x(t))$ to zero, which in turns implies that the finite time consensus condition (4.14) *is achieved.*

Now, we show that when the set of agents changes, the value of the considered non-smooth Lyapunov function (4.15) *does not change.*

Let one agent join the network at time $t^+=t+\varepsilon$ where $\varepsilon>0$ is arbitrarily small. According to (4.3)*, it joins with a state value equal to the average of its neighbours, i.e., inside the convex hull spanned by their states. Thus, we can write:*

$$\max_{i\in V(t^+)} x_i(t)=\max_{i\in V(t)} x_i(t),\qquad \min_{i\in V(t^+)} x_i(t)=\min_{i\in V(t)} x_i(t)$$

Therefore, the value of the Lyapunov function (4.15) *does not change, i.e.,* $\mathscr{V}_1(x(t^+)) = \mathscr{V}_1(x(t))$*. If, instead, an agent leaves the network, it is straightforward to verify that the Lyapunov function may decrease in value and* $\mathscr{V}_1(x(t^+)) \leq \mathscr{V}_1(x(t))$*. Now, by integrating the generalized time derivative* (4.22) *in an interval of time from which we remove all sets of instants of measure zero corresponding to changes in the network topology, or agents that join or leave the network, and considering that when these events occur they do not increase the value of function* $\mathscr{V}_1(x(t))$*, it holds:*

$$\mathscr{V}_1(x(t)) = \mathscr{V}_1(x(0)) + \int_0^t \frac{d}{dt}\left(\mathscr{V}_1(x(t))\right) dt \leq \mathscr{V}_1(x(0)) - \mu^2 t. \tag{4.23}$$

In [55] the parameter μ^2 *was estimated to be* $\mu^2 = 2\left(\frac{2\lambda}{n} - \alpha\right)$*. In this work the number of agents is time-varying, therefore we extend the same result by estimating its value as* $\mu^2 = 2(\frac{2\lambda}{n_{max}} - \alpha)$ *based on Assumption 4.3.5. Thus, by Theorem 2.4.2,* $\mathscr{V}_1(x(t))$ *converges to zero in finite time and from* (4.23) *we can compute a maximum finite transient time which is equal to* $T_1 = \left(\max_{i\in\mathscr{V}(t)} x_i(0) - \min_{i\in\mathscr{V}(t)} x_i(0)\right)/\mu^2$. ■

The next theorem proves that the dynamic consensus function $c(t)$ converges in finite time and tracks with zero error the median value $m(u(t))$ despite a switching network topology if no agent joins or leaves the network.

Theorem 4.4.2

Consider the network dynamics (4.1) along with a time-varying graph $G(t)$ with n agents satisfying Assumption 4.3.3. Let the tuning parameters, under Assumption 4.3.1, satisfy $n\Pi < \alpha < \frac{2\lambda}{n}$ and let the consensus condition (2.3) be in force starting from time $t = T$. Then, there exist $T_2 \geq T$ such that

$$c(t) \in \mathscr{M}(u(t)), \quad \forall t \geq T_2, \tag{4.24}$$

where $\mathscr{M}(u(t))$ denotes the time-varying median interval which contains the median value (4.5) of the set of reference signals and

$$T_2 \leq \frac{|c(T) - m(u(T))| + L_M(T_2)}{\frac{\alpha}{n} - \Pi} + T. \tag{4.25}$$

Proof of Theorem 4.4.2 *By assumption, the consensus condition* (2.3) *is in force for all* $t \geq T$*. Consider the Lyapunov function*

$$\mathscr{V}_2(x(t), u(t)) = |c(t) - m(u(t))|, \quad t \geq T. \tag{4.26}$$

In the following, to simplify our notation we omit the dependence of $\mathscr{V}_2(x(t), u(t))$ *from* $x(t)$ *and* $u(t)$ *and refer to it as* $\mathscr{V}_2(t)$ *instead. It is worth to note that due to* (2.3) *all* *agents hold the same state value $c(t)$, and therefore the average state value is given by $c(t)$, i.e.,*

$$c(t) = \frac{\sum_{i\in V(t)} x_i(t)}{n}, \quad t \geq T. \tag{4.27}$$

The generalized gradient of $\mathscr{V}_2(t)$ takes the form:

$$\partial\mathscr{V}_2(t) = \mathrm{SIGN}(c(t) - m(u(t))). \tag{4.28}$$

By (4.27)*, the time varying consensus function $c(t)$ obeys the discontinuous differential equation $\dot{c}(t) = \frac{\sum_{i\in V}\dot{x}_i(t)}{n}$, thus*

$$\dot{c}(t) = -\frac{\sum_{i\in V}\alpha\,\mathrm{sign}\,(x_i(t) - u_i(t))}{n} + \frac{\lambda\sum_{j\in\mathscr{N}_i(t)}\mathrm{sign}\left(x_i(t) - x_j(t)\right)}{n}. \tag{4.29}$$

Due to the symmetry of local interactions resulting from the undirected nature of the considered graph $G(t)$, it is $\sum_{i\in V}\lambda\left(\sum_{j\in\mathscr{N}_i(t)}\mathrm{sign}\left(x_i(t) - x_j(t)\right)\right) = 0$, therefore (4.29) *straightforwardly simplifies as*

$$\dot{c}(t) = -\frac{\sum_{i\in V}\alpha\cdot\,\mathrm{sign}\,(x_i(t) - u_i(t))}{n}. \tag{4.30}$$

The Filippov solutions of (4.30) *are governed by the differential inclusion [60]*

$$\dot{c}(t) \in -\frac{1}{n}\alpha\sum_{i\in V}\mathrm{SIGN}\,(x_i(t) - u_i(t))\,. \tag{4.31}$$

Let

$$I_{up}(t) = \{k \in V : x_k(t) < u_k(t)\}, \tag{4.32}$$

$$I_{down}(t) = \{k \in V : x_k(t) > u_k(t)\}, \tag{4.33}$$

$$I_{equal}(t) = \{k \in V : x_k(t) = u_k(t)\}. \tag{4.34}$$

Clearly, the sets $I_{up}(t)$, $I_{down}(t)$ and $I_{equal}(t)$ are disjoint, and their union forms the set V, thus

$$|I_{up}(t)| + |I_{down}(t|) + |I_{equal}(t)| = |V| = n \tag{4.35}$$

By definition, the next relations hold:

$$\mathrm{SIGN}\,(x_i(t) - u_i(t)) = -1, \quad \forall i \in I_{up}(t), \tag{4.36}$$

$$\mathrm{SIGN}\,(x_i(t) - u_i(t)) = 1, \quad \forall i \in I_{down}(t). \tag{4.37}$$

Therefore, by (4.36) *and* (4.37) *one manipulates* (4.31) *as follows:*

$$\dot{c}(t) \in -\frac{\alpha}{n}\left(|I_{down}(t)| - |I_{up}(t)| + \sum_{i\in I_{equal}(t)}\mathrm{SIGN}(x_i(t) - u_i(t))\right). \tag{4.38}$$

The set-valued Lie derivative of $\mathscr{V}_2(t)$ *takes the form:*

$$\widetilde{\mathscr{L}}\mathscr{V}_2(t) = \mathrm{SIGN}(c(t) - m(u(t)))\dot{c}(t) - \mathrm{SIGN}(c(t) - m(u(t)))\dot{m}(u(t)). \tag{4.39}$$

We notice that, by assumption, it is

$$x_i(t) = c(t), \qquad \forall i \in \mathscr{V}, \quad \forall t \geq T, \tag{4.40}$$

and by definition, $c(t) < u_i(t)$, $\forall i \in I_{up}(t)$ *and* $c(t) > u_i(t)$, $\forall i \in I_{down}(t)$. *Having in mind the definition of the median value, the next implication holds:*

$$|I_{down}(t)| = |I_{up}(t)| \quad \Longrightarrow \quad c(t) \in \mathscr{M}(u(t)). \tag{4.41}$$

Therefore, we concentrate on the case in which $|I_{down}(t)| \neq |I_{up}(t)|$. *When* $|I_{down}(t)| \neq |I_{up}(t)|$, *two cases may occur*

$$\textit{Case 1}: \big||I_{down}(t)| - |I_{up(t)}|\big| > |I_{equal}(t)|, \tag{4.42}$$

$$\textit{Case 2}: \big||I_{down}(t)| - |I_{up}(t)|\big| \leq |I_{equal}(t)|, \tag{4.43}$$

which will be treated separately.

Case 1. *When relation* (4.42) *is in force, it holds:*

$$\sum_{i \in I_{equal}(t)} \alpha\ \mathrm{SIGN}(x_i(t) - u_i(t)) \in [-\alpha|I_{equal}(t)|, \alpha|I_{equal}(t)|]. \tag{4.44}$$

Without loss of generality we consider the case $|I_{down}(t)| > |I_{up}(t)|$ *and thus inequality* (4.42) *becomes* $|I_{down}(t)| - |I_{up}(t)| > |I_{equal}(t)|$ *(the same derivation can be carried out for the case* $|I_{down}(t)| < |I_{up}(t)|$*). Then, we consider the value of each term in eq.* (4.38) *thus yielding the next estimate:*

$$\alpha \left| |I_{down}(t)| - |I_{up}(t)| + \sum_{i \in I_{equal}(t)} \mathrm{SIGN}(x_i(t) - u_i(t)) \right| \geq \alpha \left(|I_{down}(t)| - |I_{up}(t)| - |I_{equal}(t)| \right). \tag{4.45}$$

Denote

$$k(t) = |I_{up}(t)| + |I_{equal}(t)|. \tag{4.46}$$

By substituting eq. (4.46) *into eq.* (4.35), *it follows that:*

$$|I_{down}(t)| = n - k(t). \tag{4.47}$$

Now substituting (4.46) *into* $|I_{down}(t)| > |I_{up}(t)| + |I_{equal}(t)|$, *it holds that* $|I_{down}(t)| > k(t)$. *Since* $|I_{down}(t)|$ *and* $k(t)$ *are integer numbers, then* $|I_{down}(t)| \geq k(t) + 1$. *Thus, since the smaller lower bound of* $|I_{down}(t)|$ *occurs for n odd, it holds*

$$k(t) \leq \frac{n-1}{2}. \tag{4.48}$$

In light of Eq. (4.46) *and Eq.* (4.47)*, the right hand side of eq.* (4.45) *can be rewritten as:*

$$\alpha\left(|I_{down}(t)| - |I_{up}(t)| - |I_{equal}(t)|\right) = \alpha\left(n - 2k(t)\right). \tag{4.49}$$

Since the right hand side of Eq. (4.49) *is a decreasing function of $k(t)$, its minimum subject to the constraint in Eq.* (4.48) *is obtained when $k(t) = \frac{n-1}{2}$. Thus, it holds $\alpha\left(n - 2k(t)\right) \geq \alpha(n - 2\frac{n-1}{2}) = \alpha$ and by virtue of* (4.45)*, one derives that:*

$$\alpha\left| |I_{down}(t)| - |I_{up}(t)| + \sum_{i \in I_{equal}(t)} \text{SIGN}(x_i(t) - u_i(t)) \right| \geq \alpha. \tag{4.50}$$

Additionally, (4.42) *and* (4.44) *also imply that:*

$$\text{sign}\left(|I_{down}(t)| - |I_{up}(t)| + \sum_{i \in I_{equal}(t)} \text{SIGN}(x_i(t) - u_i(t)) \right) = \text{sign}\left(|I_{down}(t)| - |I_{up}(t)|\right).$$

By exploiting the definition of the median value, and considering (4.40) *and* (4.42) *one derives the following implications:*

$$|I_{down}(t)| < |I_{up}(t)| \iff c(t) < \inf \mathscr{M}(u(t)), \tag{4.51}$$

$$|I_{down}(t)| > |I_{up}(t)| \iff c(t) > \sup \mathscr{M}(u(t)). \tag{4.52}$$

Relations (4.51) *and* (4.52) *imply in turn that for $|I_{down}(t)| \neq |I_{up}(t)|$*

$$\text{sign}\left(|I_{down}(t)| - |I_{up}(t)|\right) = \text{sign}\left(c(t) - m(u(t))\right). \tag{4.53}$$

Therefore, under Assumption 4.3.1, by Proposition 4.3.1, and by eq.s (4.39)*,* (4.50) *and* (4.53)*, it holds $\max_{\xi \in \widetilde{\mathscr{L}\mathscr{V}_2}(t)} \xi \leq -\frac{\alpha}{n} + \Pi$. Thus, since $\frac{d}{dt}\left(\mathscr{V}_2(t)\right) \in \widetilde{\mathscr{L}\mathscr{V}_2}(t)$, unless $c(t) \in \mathscr{M}(u(t))$ (i.e., $\mathscr{V}_2(t) \leq L_M$) it holds*

$$\frac{d}{dt}\left(\mathscr{V}_2(t)\right) \leq -\frac{\alpha}{n} + \Pi. \tag{4.54}$$

Thus, in Case 1, the finite time achievement of condition $c(t) \in \mathscr{M}(u(t))$ is guaranteed by analogous developments as those made in the proof of Theorem 4.4.1, with a finite transient time satisfying (4.25)*.*

Case 2. *This case may only happen in the event that more than one agent holds the same value of their corresponding reference signal, $u_i(t)$, and additionally this value corresponds to the median value. We now prove that if condition* (4.43) *holds then $c(t) \in \mathscr{M}(u(t))$. Define*

$$k_{up}(t) = |I_{up}(t)| - n + 1, \quad k_{down}(t) = |I_{down}(t)|. \tag{4.55}$$

Thus

$$|I_{equal}(t)| = (k_{up}(t) - k_{down}(t)) - 1. \tag{4.56}$$

Relation (4.43) *yields the two inequalities:*

$$|I_{up}(t)| - |I_{down}(t)| \le |I_{equal}(t)| \quad \textit{if} \quad |I_{up}(t)| > |I_{down}(t)| \tag{4.57}$$
$$|I_{down}(t)| - |I_{up}(t)| \le |I_{equal}(t)| \quad \textit{if} \quad |I_{up}(t)| < |I_{down}(t)| \tag{4.58}$$

By substituting (4.55), *and* (4.56) *into* (4.57) *it yields*

$$|I_{up}(t)| \le \frac{n}{2} \tag{4.59}$$

Since we are investigating the case $|I_{up}(t)| > |I_{down}(t)|$, *it derives from* (4.59) *and* (4.55) *that* $k_{down}(t) < \frac{n}{2}$. *Thus, if* $|I_{up}(t)| > |I_{down}(t)|$ *then the set* $I_{equal}(t)$ *contains the index of at least one agent whose state belongs to the median interval* $\mathscr{M}(u(t))$ *(or is equal to the median value* $m(u(t))$ *if* n *is odd). It follows that, since the network is at consensus, all agents in set* $I_{equal}(t)$ *have state value equal to the median value.*

By applying similar considerations to (4.58) *one can derive the same result which implies that* $c(t) \in \mathscr{M}(u(t))$.

To evaluate the convergence time, let us write down the inequality

$$\begin{aligned}\mathscr{V}_2(t) &= \mathscr{V}_2(T) + \int_T^t \frac{d}{dt}(\mathscr{V}_2(t))\,dt \le \mathscr{V}_2(T) + \int_T^t \left(-\frac{\alpha}{n} + \Pi\right) dt \\ &= \mathscr{V}_2(T) - \left(\frac{\alpha}{n} - \Pi\right)(t - T).\end{aligned}$$

By letting $\mathscr{V}_2(t) \le L_M(t)$, *for* $t = T_2$ *it holds* $\mathscr{V}_2(T) - \left(\frac{\alpha}{n} - \Pi\right)(T_2 - T) \le L_M(T_2)$, *thus*

$$T_2 \le \frac{\mathscr{V}_2(T) - L_M(T_2)}{\left(\frac{\alpha}{n} - \Pi\right)} + T = \frac{|c(T) - m(u(T))| - L_M(T_2)}{\frac{\alpha}{n} - \Pi} + T. \tag{4.60}$$

■

Next, we consider the case where agents join and leave the network according to Assumptions 4.3.4 and 4.3.5.

Theorem 4.4.3

Consider the network dynamics (4.1) along with time-varying graph $G(t)$ satisfying Assumption 4.3.3. Assume that agents join and leave the network according to Assumptions 4.3.4 and 4.3.5. Let the tuning parameters, under Assumption 4.3.1, satisfy $n_{max}\Pi < \alpha < \frac{2\lambda}{n_{max}}$ and let the consensus condition (2.3) be in force starting from time $t = T$. Under Assumption 4.3.2, there exists T_3 such that

$$|c(t) - m(u(t))| \le B, \quad \forall t \ge T_3, \tag{4.61}$$

where $B \leq \left(\frac{\alpha}{n_{max}} - \Pi\right)\Delta\tau$ and

$$T_3 \leq \left(\frac{\max\{|c(T) - m(u(T))| - B, 0\}}{\left(\frac{\alpha}{n_{max}} - \Pi\right)\Delta\tau - B}\right)\Delta\tau + T. \tag{4.62}$$

Proof of Theorem 4.4.3 *According to Assumption 4.3.4 there exists a minimum dwell time $\Delta\tau$ between two consecutive join or leave events. Consider two instants of time, t' where $|V(t')| = n(t')$, and $t'' = t' + \Delta\tau$ where $|V(t'')| = n(t'')$. Thus for $t \in (t', t'')$ there is at most one agent that either joins or leaves the network at the isolated instant of time $\hat{t} \in (t', t'')$. Thus, we can apply the results of Theorem 4.4.2 and disregard the isolated instant $\hat{t}$ in the Lebesgue integration of the Lyapunov function. In particular, by Assumption 4.3.5 (maximum number n_{max} of agents in the network), it holds $\frac{d}{dt}(\mathscr{V}_2(t)) \leq -\frac{\alpha}{n_{max}} + \Pi$. By Assumption 4.3.4, $|n(t'') - n(t')| \leq 1$. Now, we compute the variation of $\mathscr{V}_2(t)$ at time $\hat{t}$ when the join or leave event occurs. It holds $\mathscr{V}_2(\hat{t}^+) - \mathscr{V}_2(\hat{t}) = |c(\hat{t}^+) - m(u(\hat{t}^+))| - |c(\hat{t}) - m(u(\hat{t}))|$. Since $c(\hat{t}^+) = c(\hat{t})$, because agents join with state value equal to the average of their neighbors and the consensus condition is enforced for all $t \geq T$, we can write $\mathscr{V}_2(\hat{t}^+) - \mathscr{V}_2(\hat{t}) = |c(\hat{t}) - m(u(\hat{t}^+))| - |c(\hat{t}) - m(u(\hat{t}))|$. Furthermore, by Assumption 4.3.2, it holds $|m(u(\hat{t}^+)) - m(u(\hat{t}))| \leq B$. Thus $c(\hat{t}) - m(u(\hat{t}^+)) \leq c(\hat{t}) - m(u(\hat{t})) + B$ and*

$$\begin{aligned} |\mathscr{V}_2(\hat{t}^+) - \mathscr{V}_2(\hat{t})| &\leq ||c(\hat{t}) - m(u(\hat{t})) + B| - |c(\hat{t}) - m(u(\hat{t}))|| \\ &\leq ||c(\hat{t}) - m(u(\hat{t}))| + B - |c(\hat{t}) - m(u(\hat{t}))|| \\ &= B. \end{aligned} \tag{4.63}$$

We now integrate the Lyapunov function during an interval of length equal to the dwell time $\Delta\tau$ by taking into account (4.54) *and* (4.63) *and disregarding the set of instants of time of measure zero in which graph $G(t)$ changes, according to Assumption 4.3.3. It holds*

$$\begin{aligned} \mathscr{V}_2(t' + \Delta\tau) &= \mathscr{V}_2(t') + \int_{t'}^{\hat{t}} \frac{d}{dt}(\mathscr{V}_2(t))\,dt + \mathscr{V}_2(\hat{t}^+) - \mathscr{V}_2(\hat{t}) + \int_{\hat{t}^+}^{t'+\Delta\tau} \frac{d}{dt}(\mathscr{V}_2(t))\,dt \\ &\leq \mathscr{V}_2(t') + \left(-\frac{\alpha}{n_{max}} + \Pi\right)(\hat{t} - t') + B + \left(-\frac{\alpha}{n_{max}} + \Pi\right)(t' + \Delta\tau - \hat{t}^+). \end{aligned} \tag{4.64}$$

Since $\hat{t}^+ - \hat{t} = 0$, it holds $\mathscr{V}_2(t' + \Delta\tau) \leq \mathscr{V}_2(t') - \left(\frac{\alpha}{n_{max}} - \Pi\right)\Delta\tau + B$. If we consider time $t' = T$ as the initial instant of time in which consensus is enforced, then there exists $k \in \mathbb{N}^+$ such that

$$\mathscr{V}_2(T + k\Delta\tau) \leq \mathscr{V}_2(T) - k\left(\left(\frac{\alpha}{n_{max}} - \Pi\right)\Delta\tau - B\right). \tag{4.65}$$

Thus, if $B \leq \left(\frac{\alpha}{n_{max}} - \Pi\right)\Delta\tau$ *then there exists* k' *and* $T_3 \leq k'\Delta\tau + T$ *such that by letting* $\mathscr{V}_2(T_3) \leq B$*, it holds* $\mathscr{V}_2(T) - k'\left(\left(\frac{\alpha}{n_{max}} - \Pi\right)\Delta\tau - B\right) \leq B$*, thus*

$$k' \geq \frac{\max\{\mathscr{V}_2(T) - B, 0\}}{\left(\frac{\alpha}{n_{max}} - \Pi\right)\Delta\tau - B} = \frac{\max\{|c(T) - m(u(T))| - B, 0\}}{\left(\frac{\alpha}{n_{max}} - \Pi\right)\Delta\tau - B}. \tag{4.66}$$

and $T_3 \leq k'\Delta\tau + T$. ■

Remark 4.4.1

According to Theorem 4.4.3 a procedure to tune the parameters of the proposed protocol to guarantee the achievement of dynamic consensus in finite time in the *open* scenario with bounded error is simply to choose a sufficiently large α so that $\alpha > n_{max}\Pi$, then choose a sufficiently large λ so that $\lambda > n_{max}\alpha$.

4.5 Simulation results

In this section we present two numerical tests. In the first one, we consider a network of 5 agents connected by a random graph. We choose the tuning gains of the protocol according to Theorem 4.4.1 as $\lambda = 90$, $\alpha = 8$ and the initial states of the agents are choosen uniformly at random in the interval $[0, 2]$. We considered reference signals equal to $u_i(t) = a_i sin(2\pi f_i t)$ with coefficients chosen uniformly at random in the intervals $a_i \in [0, 2]$, $f_i \in [0, 0.05]$. In Figure 4.1, the reference signals and their median value are shown.

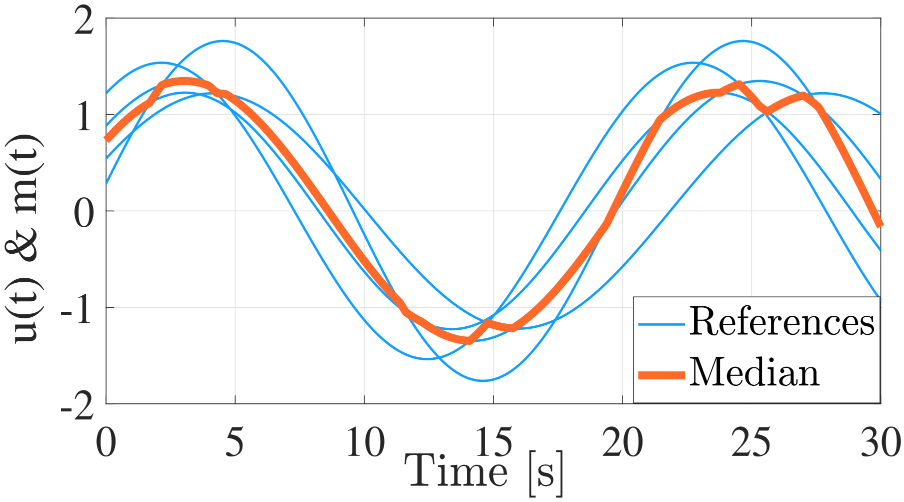

Figure 4.1: Reference signals $u_i(t)$ and their median value $m(u(t))$.

Figure 4.2 shows the consensus value of the agents, which can be seen to track, after a finite transient time (approximately 0.03 seconds, less than the analytical upper bound of $T_1 = 0.16$ seconds), the median value of the reference signals, starting at time $t = 0.6$ seconds.

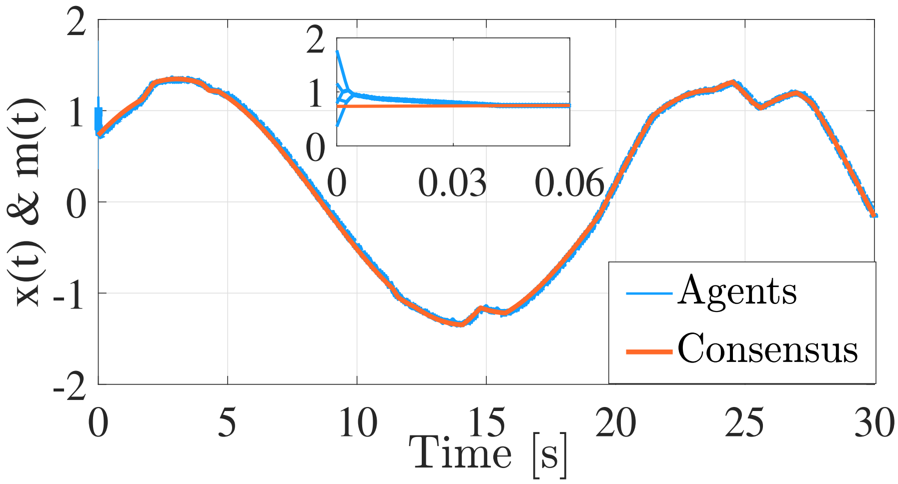

Figure 4.2: Time evolution of the network state $x(t)$ and the median value $m(u(t))$.

In the second test, we consider the scenario of an open multi-agent system. The simulation starts with 9 agents at the consensus state on the median value $m(u(t))$ of their reference signals, then at $t = 10$ an agent joins the network with initial state according to (4.3) and $u_{10}(t) = 1.3sin(2\pi 0.048t + 0.9)$. Furthermore, at $t = 18$ sec a second agent joins the network, and finally at $t = 27$ sec an agent leaves the network. It can be seen in Figures 4.3 and 4.4 that when agents join or leave the network the median value of the reference signals changes. In Figure 4.4, it is shown that the state of the agents remains at the consensus state and at bounded distance from $m(u(t))$ during the join and leave events.

4.6 Conclusions

In this chapter, we considered the dynamic consensus problem on median value in an open multi-agent system under a switching network topology. The considered protocol is shown to track with zero error in finite-time the median value of a set of time-varying reference signals with bounded variations. Furthermore, it is shown that when agents join and leave the network with an appropriate dwell time, the consensus value tracks with bounded error the median value. Future works will investigate discrete-time versions of the proposed approach and its resilience with respect to faults or malicious agents.

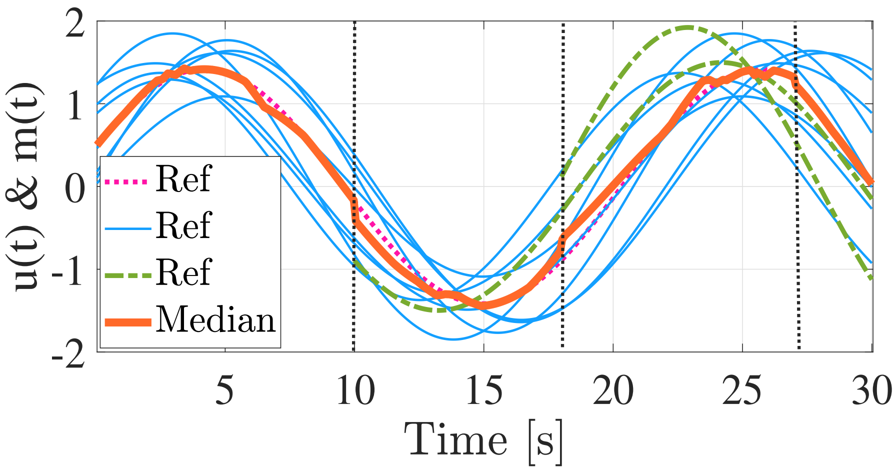

Figure 4.3: Reference signals $u_i(t)$ and their median value $m(u(t))$.

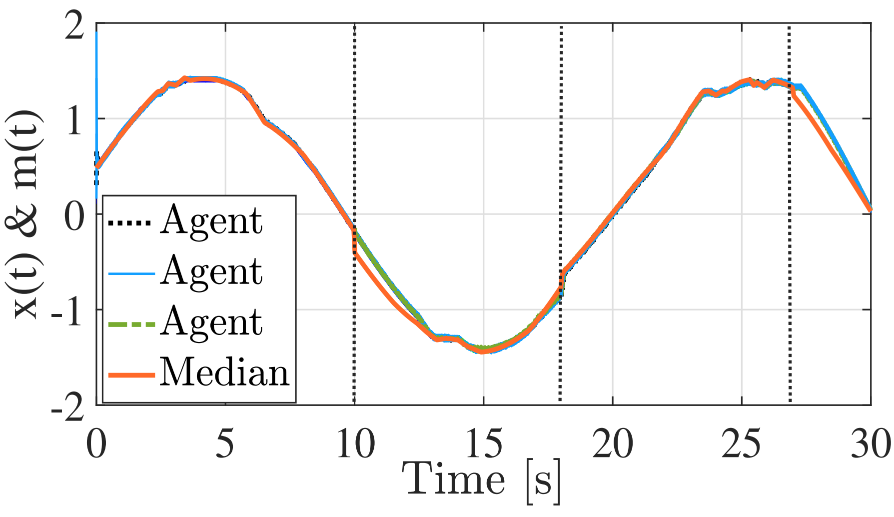

Figure 4.4: Time evolution of the network state $x(t)$ and the median value $m(u(t))$.

Chapter 5

Distributed heuristics for self-organizing k-regular random graphs

5.1 Introduction

In this chapter, we present a novel distributed algorithm to reshape any connected graph with a possible non-integer average degree into a connected approximately random k-regular graph which is independent of the degree of the initial graph. More specifically, we define some local graph transformation rules and provide a distributed implementation. In the resulting process, a random regular graph is obtained while the agents observe and modify only the local structure of the network. As such, the network achieves high expansion ratios and algebraic connectivity, which provide robustness to various structural and functional perturbations.

The outline of this chapter is as follows: In Section 5.2 we give the contributions of this chapter. In Section 5.3 we present needed background on some graph theory and definitions. In Section 5.4 we provide a distributed implementation and an analysis of the resulting dynamics. In Section 5.5 we build some simulation results to demonstrate the performance of the proposed solution. Finally, Section 5.6 concludes this chapter.

5.2 Main contributions

The robustness of a multi-agent network to perturbations such as failures and noise largely depends on the corresponding graph. In many applications, networks are desired to have well-connected interaction graphs with relatively small number of links. One family of such graphs is the random regular graphs. Thus, refs. [56–58] motived to design a decentralized protocol for transforming any connected interaction graph into a connected random regular graph. In more detail, in [56] and [57], to robustify multi-agent systems against perturbations, the authors proposed algorithms where any graph with an integer average degree, $m \in \mathbb{N}$, was transformed into a random k-regular graph while the protocol in [58], rewires any graph with a possible non-integer average degree of m into a random k-regular graph for some $k \in [m, m+2]$. However, k-regular random graph built by the aforementioned works is dependent on the average degree of the initial graph. Thus, we motivated to design a distributed scheme for transforming any connected interaction graph with a possible non-integer average degree into a connected approximately random k-regular graph which is independent of the degree of the initial graph. Hence, the network is well connected with a relatively small number of links that leads to improved robustness of networks.

5.3 Preliminaries

Let $G = (V, E(t), A(t))$ be an undirected graph, where $V = \{1, \ldots, N\}$ is a set of nodes (agents), $E(t) \subseteq \{V \times V\}$ is the time-varying set of edges, given by unordered pairs of nodes, and $A(t) = (a_{ij})_{N\times N}$ represents a weighted adjacency matrix. In detail, $a_{ii=0}$, $a_{ij} \geq 0$, and $a_{ij} > 0$ if and only if the link $(V_i, V_j) \in E(t)$. The Laplacian matrix $L = (l_{ij})_{N\times N}$ is denoted as $l_{ij} = -a_{ij}$, $i \neq j$, and $l_{ii} = \sum_{k=1}^{N} a_{ik}$ for all i. An undirected graph is connected when it has at least one node, and there is a path between every pair of nodes. A path is a sequence of nodes such that an edge exists between any two consecutive nodes in the sequence. Two nodes in an undirected graph are said to be adjacent if they are joined by an edge. We refer to the set of nodes adjacent to any $i \in V$ node, as its 1-hop neighborhood, $N_i^1(t)$, defined as:

$$N_i^1(t) = \{j | (i, j) \in E(t)\} \tag{5.1}$$

Furthermore, we define a set of neighbors of each neighbor of node i as 2-hop neighborhood, $N_i^2(t)$. For any node i, the number of nodes in its 1-hop neighborhood is called its degree, $d_i(t)$, i.e.,

$$d_i(t) = |N_i^1(t)| \tag{5.2}$$

where $|N_i^1(t)|$ denotes the cardinality of the set. Let us define $d_{\min}(G(t))$, $d_{\max}(G(t))$, and $\bar{d}(G(t))$ for any graph $G(t)$ as the minimum, the maximum, and the average degrees, respectively. Moreover, we denote the difference of the maximum and the minimum node degrees in a graph as the degree range, $f(G(t))$, i.e.,

$$f(G(t)) = |d_{\max}(G(t)) - k| + |k - d_{\min}(G(t))| \tag{5.3}$$

where $k \in \mathbb{N}$ is a desired value.

Before presenting the proposed algorithm, we provide the following definitions.

Definition 5.3.1: Algebraic connectivity

The second-smallest eigenvalue of the graph Laplacian is called as the algebraic connectivity of the graph (also known as Fiedler value or Fiedler eigenvalue), $\lambda_2(G(t))$.

Definition 5.3.2: k-regular graph [57,58]

A regular graph is a graph where each node has the same number of neighbors; i.e. every node has the same degree or valency. A regular graph with nodes of degree k is known k-regular graph or regular graph of degree k.

Definition 5.3.3: Approximately k-regular graph [57,58]

An approximately k-regular graph is a graph where all nodes except one node have a degree equal to k. As the number of nodes goes to infinity, almost every k-regular graph and approximately k-regular graph have an algebraic connectivity arbitrarily close to $\lambda_2(G) = k - 2\sqrt{k-1}$ for $k \geq 3$.

Definition 5.3.4: Random k-regular graph (approximately k-regular graph) [57,58]

A graph that is picked uniformly at random from the set of all k-regular graphs (approximately k-regular graph) with N nodes is named a random k-regular graph (approximately k-regular graphs) of order N.

5.4 Distributed formation of random regular graph

In this section, we present the proposed distributed implementation of the algorithm which transforms any connected graph into a k-random regular graph by adding, removing and moving links.

The basic ideas behind Algorithm 2 are as follows: At each iteration, each node i which is chosen randomly checks its degree $d_i(t)$ and according to the value of itself the candidate rule is picked as follows:

1. **Rule 1** ($r = r_1$): This rule is intended for cutting edges so that if $d_i(t) \geq k+1$, then i randomly finds a node from its one-hop neighborhood, $s \in N_i^1(t)$, such that the degree of node s should be greater than k. If s exists, then the link between i and s should be removed.

2. **Rule 2** $(r = r_2)$: This rule is considered to add edges so that if $d_i(t) < k$, then i chooses a node from its 2-hop neighborhood, $z \in N_i^2(t)$ with degree less than k. If

Algorithm 2: Distributed Implementation

Initialize: $G(0) = (V, E(0))$
Choose: $k \in \mathbb{N}$
At each time t **graph** $G(t) = (V, E(t))$ **is updated as follows:**
for each $i \in V$ **do**
  if $d_i(t) > k) \quad \rightarrow r = r_1$ **then**
    i picks a random $s \in N_i^1 \mid d_s(t) > k$
    if s exists **then**
      $E(t) = E(t) \setminus (i, s)$
    end if
  else if $d_i(t) < k) \quad \rightarrow r = r_2$ **then**
    i picks a $z \notin N_i^1(t), z \in N_i^2(t) \mid d_z(t) < k$
    if z exists **then**
      $E(t) = E(t) \cup (i, z)$
    end if
  end if
  if $d_i(t) \geq k) \quad \rightarrow r = r_3$ **then**
    i picks a random $j \in N_i^1(t) \mid d_j(t) \geq k$
    if j exists **then**
      i picks a random $h \in N_i^1(t) \setminus \{j\}$
      j picks a random $f \in N_j^1(t) \setminus \{i\}$
      if $((d_f(t) < k), (i, f) \notin E(t))$ **then**
        $E(t) = (E(t) \setminus \{(i, j)\}) \cup \{(i, f)\}$
      else if $((i, f) \notin E(t), (j, h) \notin E(t))$ **then**
        $E(t) = (E(t) \setminus \{(i, h), (j, f)\}) \cup \{(i, f), \{(j, h)\}$
      end if
    end if
  end if
end for

z exists, then a link is formed between i and z.

3. **Rule 3** $(r = r_3)$: This rule consists of two parts which is considered to move edges so that if $d_i(t) \geq k$, then i randomly finds a node from its one-hop neighborhood, $j \in N_i^1(t)$, such that the degree of node j should be greater than or equal to k. If j exists, then both i and j choose one neighbor, $h \neq j \in N_i^1(t)$ and $f \neq i \in N_j^1(t)$, uniformly at random. If degree node f is less than k and there is not a link between f and i, then Part 1 of Rule 3 (r_3) is executed by rewiring f to i and removing the link between j and i. Otherwise, If neither f nor h are linked to both i and j, then Part 2 of Rule 3 (r_3) is executed by rewiring h to j, f to i and removing the link between i and h, and the link between j and f.

Proposition 5.4.1

If graph $G(t)$ is connected at every time t, then during the execution of Algorithm 2, function $f(G(t))$ in Eq. (5.3) is monotonically non-increasing and if $f(G(t))$ has a value strictly greater than one, then there always exist some nodes with the value degree different from the average degree, so by applying at least one of the rules of the algorithm the value of function $f(G(t))$ can decrease.

Proof of Proposition 5.4.1 *In accordance with the algorithm at each iteration, if any graph has the function $f(G(t))$ greater than one, the following cases can be occurred:*

1. **At each instant of time if the degree of the picked node which is chosen randomly, $d_i(t)$, is greater than k, then following cases can happen:**

   (a) *Rule 1 can be executed.*

   (b) *Rule 2 cannot be executed.*

   (c) *Rule 3 (Part 1) can be executed.*

   (d) *Rule 3 (Part 2) can be executed.*

   Suppose Rule 1 is implemented. When Rule 1 is triggered, since the degree of node i and its one-hop neighborhood which is called s are strictly greater than k, the edge between these two nodes are removed. Therefore, the degree of nodes i and s is definitely reduced. If by chance the degree of one of these two nodes is the maximum degree, then $d_{\max}(G(t))$ will be decreased. On the one hand, this rule has no effect on the minimum degree. Therefore, from Eq. (5.3) function $f(G(t))$ decreases.
   Consider Rule 3 (Part 1) is executed. When this case is occurred, since the degree of node i is strictly greater than k, the degree of node j is greater or equal to k and the degree of node f is less than k, it eliminates the edge between nodes i and j and rewires a link between i and f. Thus, the degree of node i does not change, the degree of node j decreases and node f increases. By the chance If node j and node f have maximum and minimum degrees respectively, then function $f(G(t))$ will be decreased.
   Now, presume Rule 3 (Part 2) is executed. When this case happens, since the degree of node i is strictly greater than k, the degree of node j is greater or equal to k and the degree of nodes f and h is greater or equal to k, it rewires h to j, f to i, and removes the edges between i and h and between j and f. Hence, the degree of none of these nodes change which follows $d_{\max}(G(t))$ and $d_{\min}(G(t))$ remain the same value as before. Since this part of Rule 3 does not affect $d_{\max}(G(t))$ and $d_{\min}(G(t))$, then according to Eq. (5.3) function $f(G(t))$ does not change.

2. **At each instant of time if the degree of the selected node which is chosen randomly, $d_i(t)$, is less than k, then the following cases can occur:**

(a) *Rule 1 cannot be executed.*
(b) *Rule 2 can be executed.*
(c) *Rule 3 (Part 1) cannot be executed.*
(d) *Rule 3 (Part 2) cannot be executed.*

Suppose Rule 2 is executed. When this Rule is triggered, since the degree of node i and its 2-hop neighborhood which is named z is strictly less than k, a link is formed between these two nodes. Accordingly, the degree of nodes i and z is certainly increased. If by chance the degree of one of these two nodes is the minimum degree, then $d_{\min}(G(t))$ *will be increased. Since this rule has no effect on the maximum degree, subsequently according to Eq. (5.3) function* $f(G(t))$ *will be decreased.*

3. **At each instant of time if the degree of the picked node which is chosen randomly,** $d_i(t)$**, is equal to** k**, then the following cases can be occurred:**

   (a) *Rule 1 cannot be executed.*
   (b) *Rule 2 cannot be executed.*
   (c) *Rule 3 (Part 1) can be executed.*
   (d) *Rule 3 (Part 2) can be executed.*

   Consider Rule 3 is executed. As mentioned in case 1, the implementation of this rule cannot increase function $f(G(t))$.

4. **At some instant of times no rule may be executed on some nodes**

   If the degree of the selected node which is chosen randomly, $d_i(t)$ *is greater than k and none of its neighbors have a degree greater or equal to k, then no rule executes on this node which follows* $d_{\max}(G(t))$ *and* $d_{\min}(G(t))$ *do not change. Moreover, if the degree of the picked node which is chosen randomly* $d_i(t)$ *is less than k and none of its 2-hop neighbors have a degree less than k so no rule implements on this node which follows* $d_{\min}(G(t))$ *and* $d_{\max}(G(t))$ *remain the same value as before. No change in the maximum and minimum degree implies that* $f(G(t))$ *does not change. Thus, it is shown that none of the cases results in an increase in function* $f(G(t))$*. As a result, the statement of the proposition is proved.*

■

Next we discuss how Algorithm 2 ensures that an initial connected graph converges to a k-regular graph.

Theorem 5.4.1

Let $G(0) = (V, E(0))$ be an initial connected graph and it is assumed that the graph stays connected during the iterations, under the execution of the proposed algorithm after sufficient number of iterations, the graph almost surely converges to a connected approximately random k-regular graph such that $k \in \mathbb{N}$.

Proof Sketch of Theorem 5.4.1 *It is worth mentioning that in Algorithm 2, the sequence of graphs can be investigated as a Markov chain stochastic process in which the graph at time $t+1$ depends only on the graph at time t and not the graphs before t. In other words, the next graph is only a function of current graph. Let us define a Markov chain over the finite state space, $\mathbb{G}$, consisting of set of connected graphs with n nodes. The state space $\mathbb{G}$ is also considered to be the union of two disjoint sets, as follows*

$$\mathbb{G} = \begin{cases} \bar{\mathbb{G}} & \Rightarrow \{G(t) \in \mathbb{G} | d_i(G) \cong k \quad \forall i \in V\} \\ \widetilde{\mathbb{G}} & \Rightarrow \mathbb{G} \backslash \bar{\mathbb{G}} \end{cases}$$

where $\bar{\mathbb{G}}$ is the set of regular graphs whose the degree of all nodes is equal to k or just degree of a node is different from k and other nodes have degree equal to k. $\widetilde{\mathbb{G}}$ represents the set of non-regular graphs.

We are now ready to prove, each connected graph $G(t) \in \widetilde{\mathbb{G}}$ converges to the set of k-regular graphs, $\bar{\mathbb{G}}$, and remains in this set. Let $G(0) = (V, E(0)) \in \widetilde{\mathbb{G}}$ be an initial connected graph which is assumed to remain connected during the iterations with a certain value of $f(G(t))$ greater than one. By executing the proposed algorithm the following two cases can be occurred on the graph:

1. *The maximum or/and minimum degree of the graph changes*

2. *The maximum or/and minimum degree of the graph does not change*

At the beginning process since function $f(G(t))$ is greater than one, then there exists at least one node which can be picked randomly by the proposed algorithm, and related rule can be executed in accordance with its degree. This process can be done for each node $i \in V$. As proved in Proposition 5.4.1, applying the proposed algorithm on the graph $G(t)$ leads the maximum and minimum degree of the graph change to become close to k, which results in reducing function $f(G(t))$ in the value such that at time $t+1$ function $f(G(t+1))$ is less than $f(G(t))$ at time t. If there is no node and the graph does not belong to $\bar{\mathbb{G}}$, then Rule 3 (Part 2) can be executed for sufficient number of times such that at some points, nodes can be found that once selected reduce the value of $f(G(t))$. In fact, reducing function $f(G(t))$ via the proposed protocol continues until one of the following cases can be occurred.

1. **Minimum degree is reached k after sufficient number of iterations, but maximum degree is still greater than k.**

   In this case, the graph has definitely one of the following structures:

(a) **The degree of a set of nodes are greater than k and other nodes have the degree equal to k.**

Let us consider the graph $G(t)$ has many nodes with degree greater than k and less than k. As mentioned before, the maximum degree and minimum degree can become closer to k via execution of the proposed algorithm. Now, suppose minimum degree is reached k after sufficient number of iterations, but maximum degree is still greater than k. In this case there are one or more nodes whose their degrees are greater than k. If there is a number of nodes of degree greater than k, then Rule 1 and Rule 3 (Part 2) just can be executed which follows the degree of those nodes decrease until the degree of two nodes are still greater than k while other nodes have the degree equal to k. In this state, since two nodes have degree greater than k, links can remove or move which results in the degree of these two nodes (if these two nodes have the same degree greater than k) or one of these two nodes (if these two nodes have different degrees greater than k) be equal to k as the other nodes.Thereafter, there will be no node that can change maximum and minimum degree of graph, it means that the algorithm is stuck and the value of the function $f(G(t))$ does not change.Therefore, graph coverages to a connected random k-regular or approximately k-regular graph.

(b) **The degree of an agent is greater than k and other nodes have the degree equal to k.**

Assume the graph $G(t)$ has many nodes with degree greater than k and less than k. As mentioned before, the maximum degree and minimum degree can become closer to k via execution of the proposed algorithm. Now, suppose minimum degree is reached k after sufficient number of iterations, but maximum degree is still greater than k. In this case there are certainly one or more nodes whose their degrees are greater than k. If there is only one node of degree greater than k, then Rule 3 (Part 2) just can be executed that leads links can just be moved. Therefore, the degree of no node changes. Since the number of nodes are possibly large and only degree of one node is different from k, the graph is approximately a k-regular graph.

2. **Maximum degree k is reached after sufficient number of iterations, but minimum degree is still less than k.**

In this case, the graph has definitely one of the following structures:

(a) **The degree of a number of nodes are less than k and other nodes have the degree equal to k.**

Let us assume that the graph $G(t)$ has many nodes with degree greater than k and less than k. As mentioned before, the maximum degree and minimum degree can become closer to k via execution of the proposed algorithm. Now, suppose maximum degree is reached k after sufficient number of iterations, *but minimum degree is still less than k. In this case there are one or more nodes whose their degrees are less than k. If there are a number of nodes of degree less than k, then just Rule 2 and Rule 3 can be executed which follows the degree of those nodes increases until the degree of two nodes are less than k and other nodes have the degree equal to k. In this state, according to the proposed algorithm Rule 2 and Rule 3 can be executed that leads to the degree of all nodes is equal to k (if these two nodes have the same degree less than k) or the degree of an agent is $k-1$ and other agents have degree equal to k (if these two nodes have different degrees less than k). Therefore, graph surely reaches a k-regular graph or approximately a k-regular graph.*

(b) **The degree of an agent is less than k and other nodes have the degree equal to k.**

Let us consider the graph $G(t)$ has many nodes with degree greater than k and less than k. As explained before, the maximum degree and minimum degree can become closer to k via execution of the proposed algorithm. Now, suppose maximum degree is reached k after sufficient number of iterations, but minimum degree is still less than k. In this case there are one or more nodes whose their degrees are less than k. If there is only one node of degree less than k, then firstly just Rule 3 can be executed. If Rule 3 (Part 1) is executed, It means that the node with degree less than k is node f and nodes i and j have degree equal to k, then it may result two nodes or one node with degree less than k, and other nodes have degree equal to k. In this state, according to the proposed Algorithm 2, Rule 2 and Rule 3 can be executed that leads to the degree of all nodes is equal to k (if these two nodes have the same degree less than k) or the degree of an agent is $k-1$ and other agents have degree equal to k (if these two nodes have the different degree less than k). Therefore, graph surely reaches a k-regular graph or approximately a k-regular graph.

3. **Maximum degree and minimum degree reach k simultaneously after sufficient number of iterations.**

   In this case, the graph has definitely the following structure:

(a) **The degree of two nodes are equal to $k+1$ and $k-1$ respectively, and other nodes have the degree equal to k.**

In this state, just Rule 3 can be executed. If Rule 3 (Part 1) is executed, then it means that Node f has degree equal to $k-1$ and the degree of Node j is $k+1$ which implies the degree of these two nodes is equal to k as the other nodes. Therefore, graph converges to a k-regular graph.

■

Proposition 5.4.2

By executing the Algorithm 2, the graph $G(t)$ cannot get stuck, if the maximum and minimum degree of the graph is different from k. Therefore, the maximum and minim degree of the graph will change to reach to k.

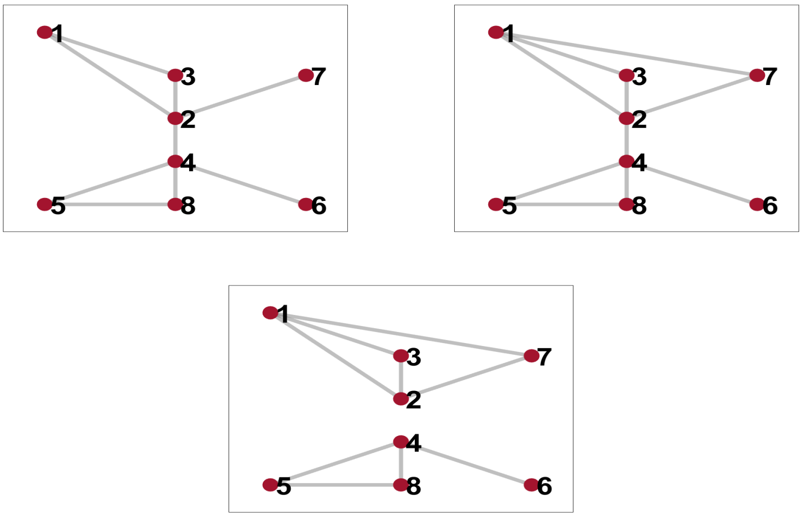

Figure 5.1: (Left) Initial interaction graph with 8 nodes and $k = 3$; (Right) The graph is updated when Rule 2 is implemented and an edge is formed between nodes 1 and 7; (Down) The graph is disconnected when Rule 1 is executed and the link between nodes 2 and 4 is removed.

Example 5.4.1

It should be noted that the execution of Algorithm 2 can disconnect any graph with non-zero probability if the initial graph has few nodes and choose a small k. The following example shows that the execution of Algorithm 2 graph can be disconnected.

In Fig. 5.1 (Left), we consider the number of agents to be 8 and $k = 3$. Then, by executing of the algorithm, the following results hold:

1. When $i = 1$, the degree of node 1 is less than k, then according to Algorithm 2, Rule 2 ($r = r_2$) should be executed, which provides a link between Node 1 and one of its 2-hop neighbourhoods with degree less than k (node 7). This step is shown in Fig. 5.1 (Right) .
2. For $i = 2$, the degree of node 2 is more than k, so Rule 1 ($r = r_1$) is executed. In accordance with this rule, i finds a node from its one-hop neighborhood such that the degree of node should be greater than k. Thus, node 4 is

selected and the link between node 2 and node 4 is removed that leads the graph will be disconnected. This process is depicted in Fig. 5.1 (Down).

Conjecture 5.4.1

If the graph has high algebraic connectivity and the number of nodes and desired k are sufficiently high, then with high probability the graph connectivity is maintained under the algorithm 2. In fact, by giving a certain number of nodes possibly large as k goes to infinity, the probability of getting the graph disconnected goes to zero.

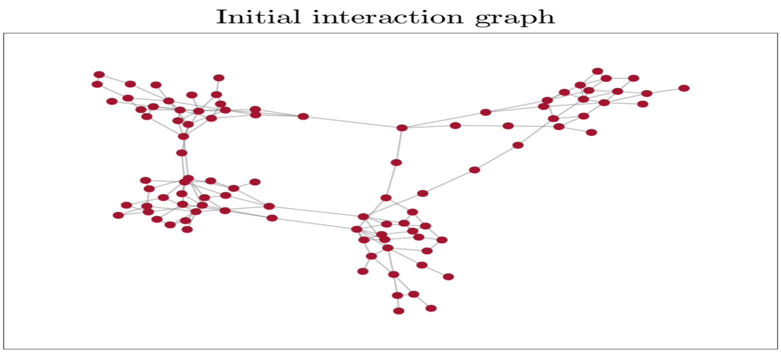

Figure 5.2: Initial interaction graph with an average degree of 3.1.

5.5 Simulation studies

In this section, we present some simulation results to evaluate the effectiveness of our proposed algorithm and the algorithm proposed in [58]. The scenarios under test consists of a connected initial graph $G(0)$ with $N = 100$ agents and $\bar{d}(G(0) = 3.1$ which is depicted in Fig. 5.2 .

In the proposed distributed protocol, we consider $k = 4$ which leads the graph converges to a connected 4-regular graph and since $\bar{d}(G(0)) = 3.1$ the graph is expected to become either a random 4-regular graph or a random 5-regular graph when the proposed algorithm in [58] is used. The obtained results are shown in Figs. 5.3 -5.9. It is clear from Fig.5.3 and Fig.5.4 that the initial interaction graph reaches a random 4-regular graph by the proposed algorithm 2 and algorithm in [58].

Time evolution of the proposed algorithm 2 average degree and average degree in [58] are shown in Fig. 5.5. It is evident from Fig. 5.5 that the proposed protocols in this chapter and in [58] guarantee the achievement of a connected 4-regular graph. However, these results show that the algorithm presented in this work is much faster

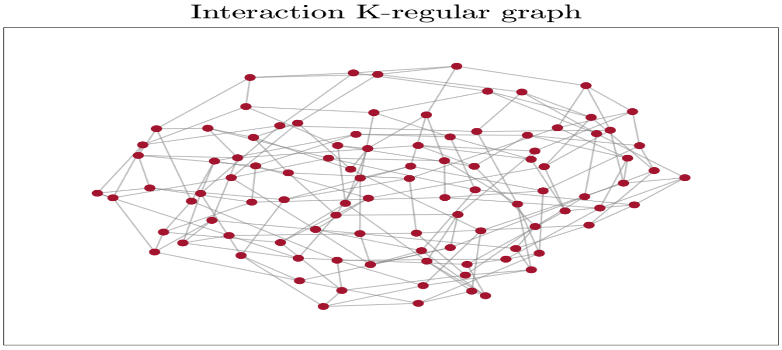

Figure 5.3: Random 4-regular graph generated by the proposed algorithm

.

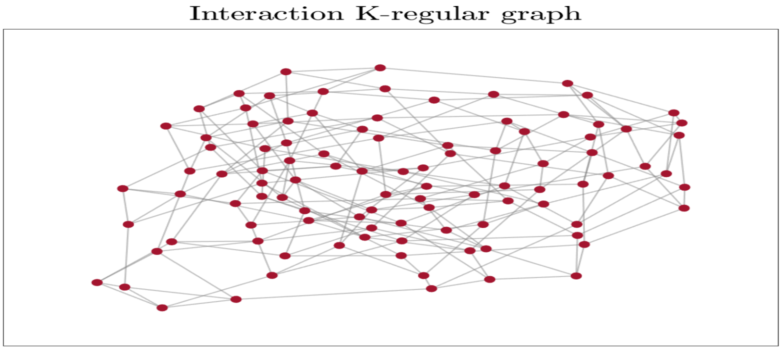

Figure 5.4: Random 4-regular graph generated in [58].

than the algorithm proposed in [58]. Fig. 5.6 illustrates time evolution of degree range in this chapter and in [58]. It can be seen from these figures that function $f(G(t))$ is not increasing during execution of the algorithm which is described in Algorithm 2 while this function is incremental at some time steps using the proposed algorithm in [58]. The time evolution of the algebraic connectivity throughout this simulation is depicted in Fig. 5.7. As predicted from random 4-regular graphs, the algebraic connectivity converges to least $k-2\sqrt{k-1} = 4-2\sqrt{3-1}$ by algorithm presented in this chapter and the proposed algorithm in [58]. Finally, time evolution of the maximum degree and minimum degree under Algorithm 2 and the proposed algorithm in [58] are displayed in Fig. 5.8 and Fig.5.9

5.6 Conclusions

In this chapter, we presented a novel distributed algorithm to transform any connected graph into a connected approximately random k-regular graph. The obtained network

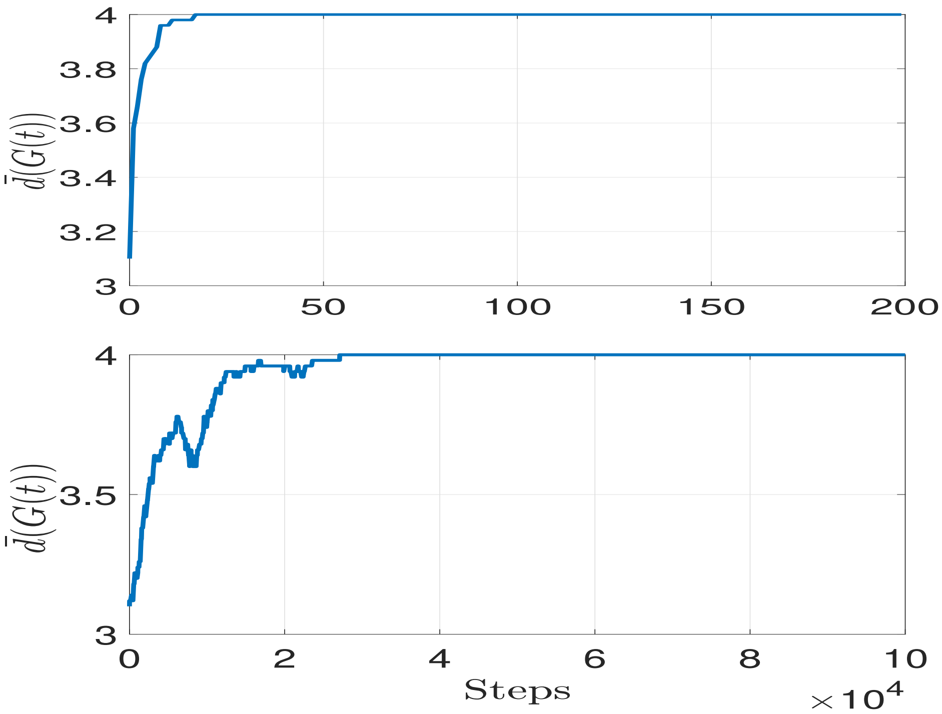

Figure 5.5: (Up) Time evolution of the proposed algorithm average degree; (Down) Time evolution of average degree in [58].

under the proposed algorithm was independent of the degree of the initial graph. Moreover, the robustness of the obtained network to perturbations such as failures, noise, or malicious attacks improved by locally adding or removing some edges.

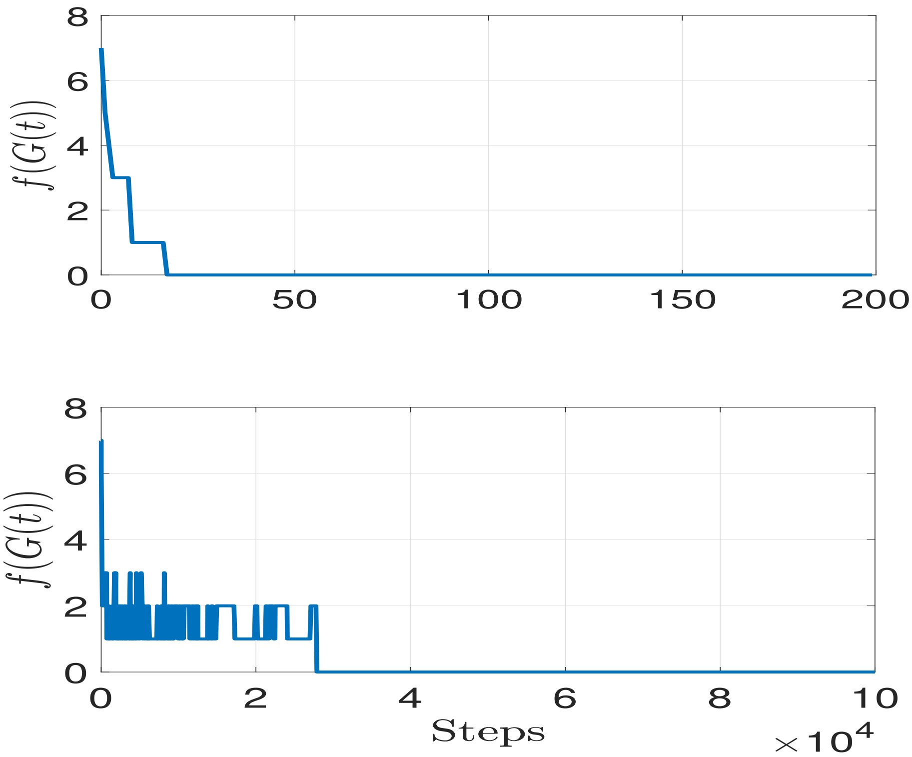

Figure 5.6: Up) Time evolution of the proposed algorithm degree range; (Down) Time evolution of degree range in [58].

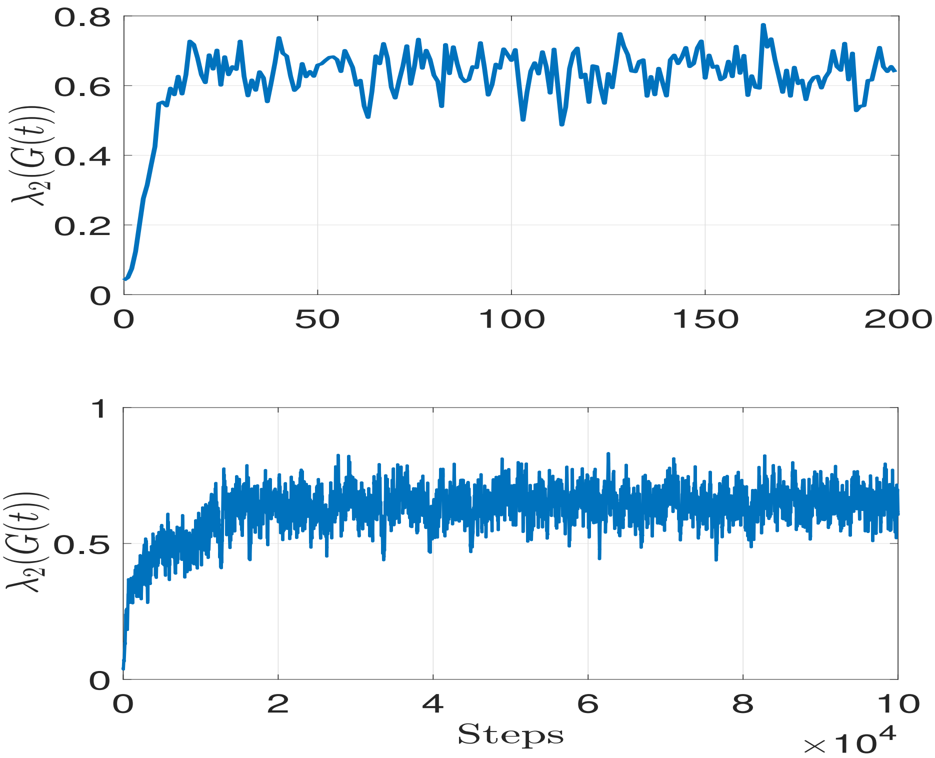

Figure 5.7: Up) Time evolution of the proposed algorithm algebraic connectivity; (Down) Time evolution of algebraic connectivity in [58].

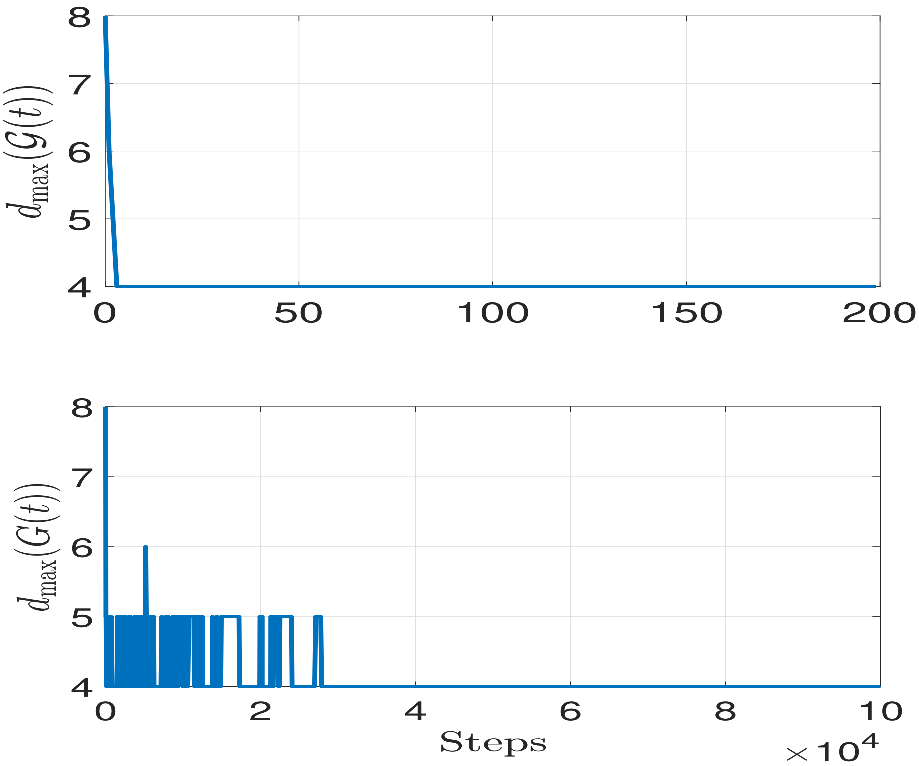

Figure 5.8: Up) Time evolution of the proposed algorithm maximum degree; (Down) Time evolution of maximum degree in [58].

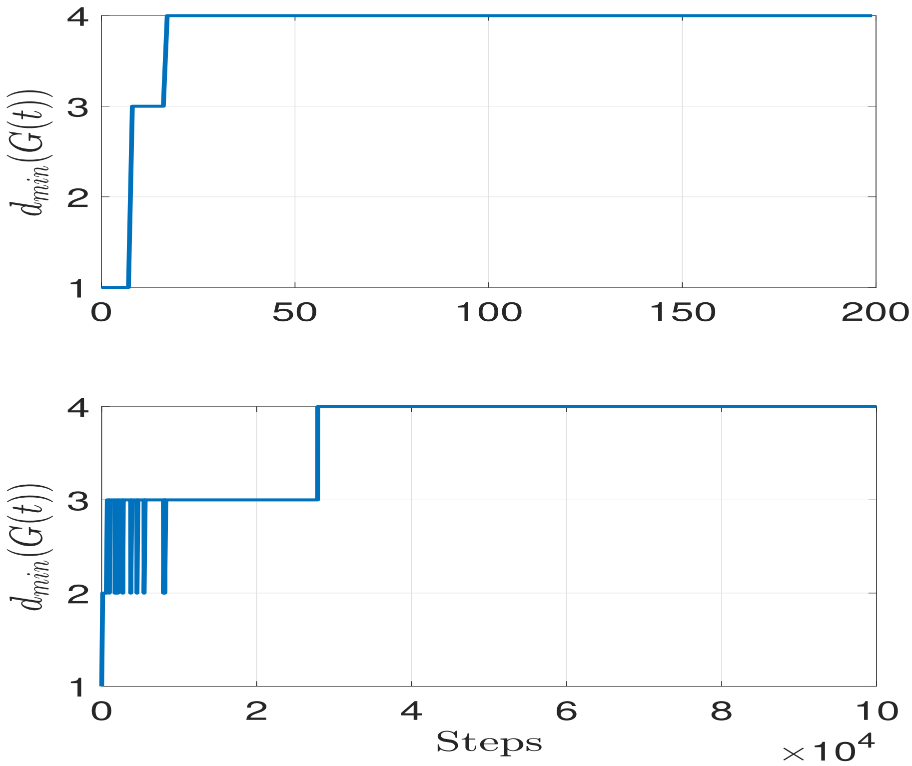

Figure 5.9: Up) Time evolution of the proposed algorithm minimum degree; (Down) Time evolution of minimum degree in [58].

Chapter 6

Conclusion and future work

6.1 Conclusion

In this concluding remarks, we review the contributions of this thesis. This thesis mainly dealt with the problem of coordination of open multi-agent systems. To solve the problem, two novel different consensus algorithms were presented in this dissertation. In Chapter 3, we studied the average discrete time consensus problem in an open multiagent system under a switching network topology. This problem was solved by designing a novel distributed open average consensus protocol. By using Lyapunov analysis We proved that when agents join or leave the network, the time-varying average of the initial states can be tracked with a bounded error by the MAS with no bias, i.e., if agents stop joining or leaving the network, it converges exactly to the average of the initial states. We also provided numerical simulations to verify the performance of the protocol. Whereas in Chapter 4, we focused on the problem of dynamic consensus of a network of agents corresponding to continuous-time systems wherein the state variables of the agents track with zero error the median value of a set of time-varying reference signals given as input to the agents under a time-varying, undirected network topology. Furthermore, we analysed the performance of the protocol in the structure of open networks by considering join and leave events, i.e., the scenario where agents may join and leave the network during the protocol execution. The finite-time convergence properties and tracking error of the considered protocol were characterized in the case of inputs with bounded variations. We employed non-smooth Lyapunov analysis to provide convergence guarantees and simple tuning rules to adjust the algorithm parameters. Finally, we built numerical simulations to corroborate the theoretical results. Besides of the coordination of open multi-agent systems problem, in Chapter 5 we designed a distributed algorithm consisting of rules for cutting, adding, and moving edges to transform any connected interaction graph with a possible non-integer average degree into a connected approximately random k-regular graph. It concluded that such graphs have the algebraic connectivity bounded away from zero regardless of the network size. Hence, it is not possible to disconnect a large part of the resulting graph by removing a small number of nodes or edges. Furthermore, by improving the algebraic connectivity, the proposed method enhances the robustness of networks to perturbations such as failures, noise, or malicious attacks. Some simulation results were also presented in the chapter to demonstrate the performance of the proposed method,

6.2 Future work

The future research directions can be divided into two parts: (i) extension of consensus approaches in the framework of open multi-agent systems (ii) extension of self-organizing k-regular random graphs.

One possible idea for the proposed average consensus is to consider the problem of dynamic average consensus in the open scenario, in which the state variables of the agents can track the average value of a set of time-varying reference signals given as input to the agents. Another idea is to focus on the addition of resilience concerning faults or malicious agents to the proposed method. A possible extension for the current median consensus is to investigate the discrete-time version of the proposed approach. The next exciting extension would be to remove Assumption 4.3.4 and consider that more than one agent can either join and leave the network at the isolated instant time. Note that in this case, we need to improve the proof of Theorem 4.3.3. Subsequent activities will also be targeted to relaxing the assumed restrictions on the communication topology by covering, e.g., directed and possibly delayed communications.

As for the extension of self-organizing k-regular random graphs, one possibility would be to analyse the performance of the protocol in the framework of open multiagent systems by proposing join and leave mechanisms, i.e., the scenario where agents may join and leave the network during the protocol execution. Another possible step could be to extend the proposed algorithm for directed graphs.

Bibliography

[1] Z. Li, Z. Duan, G. Chen, and L. Huang, "Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 213–224, 2009.

[2] D. Gu and Z. Wang, "Leader–follower flocking: algorithms and experiments," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1211–1219, 2009.

[3] R. R. Nair, L. Behera, and S. Kumar, "Event-triggered finite-time integral sliding mode controller for consensus-based formation of multirobot systems with disturbances," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 1, pp. 39–47, 2017.

[4] Z. Feng, G. Hu, Y. Sun, and J. Soon, "An overview of collaborative robotic manipulation in multi-robot systems," *Annual Reviews in Control*, vol. 49, pp. 113–127, 2020.

[5] S. Izumi, S.-I. Azuma, and T. Sugie, "Analysis and design of multi-agent systems in spatial frequency domain: Application to distributed spatial filtering in sensor networks," *IEEE Access*, vol. 8, pp. 34 909–34 918, 2020.

[6] M. A. B. Brasil, B. Bösch, F. R. Wagner, and E. P. de Freitas, "Performance comparison of multi-agent middleware platforms for wireless sensor networks," *IEEE Sensors Journal*, vol. 18, no. 7, pp. 3039–3049, 2018.

[7] M. Gholami, A. Pilloni, A. Pisano, Z. A. S. Dashti, and E. Usai, "Robust consensus-based secondary voltage restoration of inverter-based islanded microgrids with delayed communications," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 811–816.

[8] K. Cai and H. Ishii, "Average consensus on arbitrary strongly connected digraphs with time-varying topologies," *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 1066–1071, 2014.

[9] C. N. Hadjicostis, N. H. Vaidya, and A. D. Domínguez-García, "Robust distributed average consensus via exchange of running sums," *IEEE Transactions on Automatic Control*, vol. 61, no. 6, pp. 1492–1507, 2016.

[10] M. Franceschelli, A. Giua, and A. Pisano, "Finite-time consensus on the median value with robustness properties," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1652–1667, 2017.

[11] M. M. Baharloo, J. Arabneydi, and A. G. Aghdam, "Minmax mean-field team approach for a leader–follower network: A saddle-point strategy," *IEEE Control Systems Letters*, vol. 4, no. 1, pp. 121–126, 2019.

[12] J. Liu and J. Huang, "A spectral property of a graph matrix and its application to the leader-following consensus of discrete-time multiagent systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 6, pp. 2583–2589, 2018.

[13] M. Franceschelli, A. Pisano, A. Giua, and E. Usai, "Finite-time consensus with disturbance rejection by discontinuous local interactions in directed graphs," *IEEE transactions on Automatic Control*, vol. 60, no. 4, pp. 1133–1138, 2014.

[14] M. Gholami, M. Hajimani, Z. A. Z. S. Dashti, and A. Pisano, "Distributed robust finite-time non-linear consensus protocol for high-order multi-agent systems via coupled sliding mode control," in *2019 6th International Conference on Control, Instrumentation and Automation (ICCIA)*. IEEE, 2019, pp. 1–6.

[15] M. Zhu and S. Martínez, "Discrete-time dynamic average consensus," *Automatica*, vol. 46, no. 2, pp. 322–329, 2010.

[16] J. M. Hendrickx and S. Martin, "Open multi-agent systems: Gossiping with deterministic arrivals and departures," in *54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2016, pp. 1094–1101.

[17] ——, "Open multi-agent systems: Gossiping with random arrivals and departures," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 763–768.

[18] M. Abdelrahim, J. M. Hendrickx, and W. P. M. H. Heemels, "Max-consensus in open multi-agent systems with gossip interactions," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 4753–4758.

[19] M. Franceschelli and P. Frasca, "Stability of open multi-agent systems and applications to dynamic consensus," *IEEE Transactions on Automatic Control*, vol. 66, no. 5, pp. 2326–2331, 2020.

[20] ——, "Proportional dynamic consensus in open multi-agent systems," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 900–905.

[21] V. S. Varma, I. C. Morarescu, and D. Nesic, "Open multi-agent systems with discrete states and stochastic interactions," *IEEE Control Systems Letters*, vol. 2, no. 3, pp. 375–380, July 2018.

[22] Y.-G. Hsieh, F. Iutzeler, J. Malick, and P. Mertikopoulos, "Optimization in open networks via dual averaging," *arXiv preprint arXiv:2105.13348*, 2021.

[23] J. M. Hendrickx and M. G. Rabbat, "Stability of decentralized gradient descent in open multi-agent systems," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 4885–4890.

[24] M. Gholami, A. Pilloni, A. Pisano, and E. Usai, "Robust distributed secondary voltage restoration control of ac microgrids under multiple communication delays," *Energies*, vol. 14, no. 4, p. 1165, 2021.

[25] M. Franceschelli and A. Gasparri, "Multi-stage discrete time and randomized dynamic average consensus," *Automatica*, vol. 99, pp. 69–81, 2019.

[26] R. Carli and S. Zampieri, "Network clock synchronization based on the second-order linear consensus algorithm," *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 409–422, 2013.

[27] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on automatic control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[28] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control systems magazine*, vol. 27, no. 2, pp. 71–82, 2007.

[29] F. Garin and L. Schenato, "A survey on distributed estimation and control applications using linear consensus algorithms," in *Networked control systems*. Springer, 2010, pp. 75–107.

[30] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters*, vol. 53, no. 1, pp. 65–78, 2004.

[31] Y. Xie and Z. Lin, "Global optimal consensus for multi-agent systems with bounded controls," *Systems & Control Letters*, vol. 102, pp. 104–111, 2017.

[32] A. I. Rikos, T. Charalambous, and C. N. Hadjicostis, "Distributed weight balancing over digraphs," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 2, pp. 190–201, 2014.

[33] B. Gharesifard and J. Cortés, "Distributed strategies for generating weight-balanced and doubly stochastic digraphs," *European Journal of Control*, vol. 18, no. 6, pp. 539–557, 2012.

[34] T. C. Aysal, M. E. Yildiz, A. D. Sarwate, and A. Scaglione, “Broadcast gossip algorithms for consensus,” *IEEE Transactions on Signal processing*, vol. 57, no. 7, pp. 2748–2761, 2009.

[35] C. Nowzari, E. Garcia, and J. Cortés, “Event-triggered communication and control of networked systems for multi-agent consensus,” *Automatica*, vol. 105, pp. 1–27, 2019.

[36] D. Spanos, R. Olfati-Saber, and R. Murray, “Dynamic consensus on mobile networks,” in *IFAC World Congress*, 2005, pp. 1–7.

[37] S. Nosrati, M. Shafiee, and M. B. Menhaj, “Dynamic average consensus via nonlinear protocols,” *Automatica*, vol. 48, no. 9, pp. 2262–2270, 2012.

[38] R. A. Freeman, P. Yang, and K. M. Lynch, “Stability and convergence properties of dynamic average consensus estimators,” in *45th IEEE Conference on Decision and Control*, 2006, pp. 338–343.

[39] S. S. Kia, J. Cortés, and S. Martinez, “Dynamic average consensus under limited control authority and privacy requirements,” *International Journal of Robust and Nonlinear Control*, vol. 25, no. 13, pp. 1941–1966, 2015.

[40] C. Hadjicostis, A. Domínguez-Garcia, and N.Vaidya, “Resilient average consensus in the presence of heterogeneous packet dropping links,” in *51th IEEE Conference on Decision and Control*, 2012, pp. 106–111.

[41] A. Dominguez-Garcia and C. Hadjicostis, “Distributed strategies for average consensus in directed graphs,” in *50th IEEE Conference on Decision and Control*, 2011, pp. 2124–2129.

[42] N. E. Manitara and C. N. Hadjicostis, “Distributed stopping for average consensus in undirected graphs via event-triggered strategies,” *Automatica*, vol. 70, pp. 121–127, 2016.

[43] B. Paden and S. Sastry, “A calculus for computing Filippov’s differential inclusion with application to the variable structure control of robot manipulators,” *IEEE Transactions on Circuits and Systems*, vol. 34, no. 1, pp. 73–82, 1987.

[44] J. Cortés, “Discontinuous dynamical systems,” *IEEE Control Systems Magazine*, vol. 28, no. 3, pp. 36–73, 2008.

[45] ——, “Finite-time convergent gradient flows with applications to network consensus,” *Automatica*, vol. 42, no. 11, pp. 1993–2000, 2006.

[46] F. Chen, Y. Cao, and W. Ren, “Distributed average tracking of multiple time-varying reference signals with bounded derivatives,” *IEEE Transactions on Automatic Control*, vol. 57, no. 12, pp. 3169–3174, 2012.

[47] J. George, R. A. Freeman, and K. M. Lynch, "Robust dynamic average consensus algorithm for signals with bounded derivatives," in *American Control Conference (ACC)*, 2017, pp. 352–357.

[48] S. Rahili and W. Ren, "Heterogeneous distributed average tracking using nonsmooth algorithms," in *American Control Conference (ACC)*, 2017, pp. 691–696.

[49] L. Wang and F. Xiao, "Finite-time consensus problems for networks of dynamic agents," *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 950–955, 2010.

[50] P. Menon and C. Edwards, "A discontinuous protocol design for finite-time average consensus," in *IEEE Conference on Control Applications*, 2010, pp. 2029–2034.

[51] K. Xu, L. Gao, F. Chen, C. Li, and Q. Xuan, "Robust finite-time dynamic average consensus with exponential convergence rates," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 7, pp. 2578–2582, 2021.

[52] S. Rao and D. Ghose, "Sliding mode control-based algorithms for consensus in connected swarms," *International Journal of Control*, vol. 84, no. 9, pp. 1477–1490, 2011.

[53] Y. Cao and W. Ren, "Finite-time consensus for multi-agent networks with unknown inherent nonlinear dynamics," *Automatica*, vol. 50, no. 10, pp. 2648 – 2656, 2014.

[54] A. Pilloni, A. Pisano, M. Franceschelli, and E. Usai, "Robust distributed consensus on the median value for networks of heterogeneously perturbed agents," in *IEEE 55th Conference on Decision and Control (CDC)*, 2016, pp. 6952–6957.

[55] M. Franceschelli, A. Giua, and A. Pisano, "Finite-time consensus on the median value with robustness properties," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1652–1667, 2017.

[56] A. Y. Yazıcıoğlu, M. Egerstedt, and J. S. Shamma, "Decentralized degree regularization for multi-agent networks," in *52nd IEEE Conference on Decision and Control*. IEEE, 2013, pp. 7498–7503.

[57] ——, "Decentralized formation of random regular graphs for robust multi-agent networks," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 595–600.

[58] ——, "Formation of robust multi-agent networks through self-organizing random regular graphs," *IEEE Transactions on Network Science and Engineering*, vol. 2, no. 4, pp. 139–151, 2015.

[59] A. Filippov, *Differential Equations with Discontinuous Righthand Sides,*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1988.

[60] F. Clarke, *Optimization and Nonsmooth Analysis*. Wiley & Sons, New York, 1983.

[61] F. Bullo, *Lectures on Network Systems*, 1st ed., 2019.

[62] R. Olfati-Saber, J. A. Fax, and R. M. Murray, “Consensus and cooperation in networked multi-agent systems,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[63] *Boston Dynamics. Changing your idea of what robots can do.*, 2021 (accessed March 24, 2021). [Online]. Available: https://www.bostondynamics.com/

[64] E. Saerens, R. Furnémont, T. Verstraten, P. L. García, S. Crispel, V. Ducastel, B. Vanderborght, and D. Lefeber, “Scaling laws of compliant elements for high energy storage capacity in robotics,” *Mechanism and Machine Theory*, vol. 139, pp. 482–505, 2019.

[65] *High Torque Series Brushless DC Motors*. Emoteq Inc., 2018, [PDF file].

[66] H. J. Bidgoly, M. N. Ahmadabadi, and M. R. Zakerzadeh, “Design and modeling of a compact rotational nonlinear spring,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4356–4361.

[67] L. L. Howell, “Compliant mechanisms,” in *21st Century Kinematics*. Springer, 2013, pp. 189–216.

[68] S. Bernhardt, “Cable guide mechanism for constant tension reel,” Jan. 29 1963, uS Patent 3,075,724.

[69] E. Hazan, *Introduction to Online Convex Optimization*, 2016, vol. 2.