UC Riverside
UC Riverside Electronic Theses and Dissertations

Title

Communication-Reduced Distributed Control and Optimization of Multi-Agent Systems

Permalink

https://escholarship.org/uc/item/838849ch

Author

Ding, Yong

Publication Date

2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
RIVERSIDE

Communication-Reduced Distributed Control and Optimization of
Multi-Agent Systems

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering

by

Yong Ding

June 2022

Dissertation Committee:

Dr. Wei Ren, Chairperson
Dr. Fabio Pasqualetti
Dr. Jay Farrell

The Dissertation of Yong Ding is approved:

_____

_____

_____
Committee Chairperson

University of California, Riverside

Acknowledgments

I would like to express my gratitude to my advisor, Dr. Wei Ren, for the great opportunity to work with him. I would like to thank Dr. Yu Zhao and Dr. Hanlei Wang for their insightful comments on my research. I would also like to thank all my lab colleagues and friends in UCR for their help in my life and research. I would like to thank my family for their support and encouragement.

To my parents for all the support.

ABSTRACT OF THE DISSERTATION

Communication-Reduced Distributed Control and Optimization of
Multi-Agent Systems

by

Yong Ding

Doctor of Philosophy, Graduate Program in Electrical Engineering
University of California, Riverside, June 2022
Dr. Wei Ren, Chairperson

This dissertation proposes communication-reduced solutions to the containment control, distributed average tracking and distributed time-varying optimization problems of multi-agent systems.

The objective of containment control in multi-agent systems is to design control algorithms for the followers to converge to the convex hull spanned by the leaders. Sampled-data based containment control algorithms are suitable for the cases where the power supply and sensing capacity are limited, due to their low-cost and energy-saving features resulting from discrete sensing and interactions. In addition, sampled-data control has advantages in performance, price and generality. On the other hand, when the agents have double-integrator dynamics and the leaders are dynamic with nonzero inputs, the existing algorithms are not directly applicable in a sampled-data setting. To this end, this dissertation proposes a sampled-data based containment control algorithm for a group of double-integrator agents with dynamic leaders with nonzero inputs under directed communication networks. By applying the proposed containment control algorithm, the followers

converge to the convex hull spanned by the dynamic leaders with bounded position and velocity containment control errors, and the ultimate bound of the overall containment error is proportional to the sampling period.

In the distributed average tracking problem, each agent uses local information to track the average of individual reference signals. In some practical applications, velocity measurements may be unavailable due to technology and space limitations, and it is also usually less accurate and more expensive to implement. Before deriving the event-triggered approach, we first present a base algorithm without using velocity measurements, which sets the stage for the development of the event-triggered algorithm. The base algorithm has an advantage over the existing related works in the senses that there is no global information requirement for parameter design. Building on the base algorithm, we present an event-triggered algorithm that further removes continuous communication requirement and is free of Zeno behavior. It is suitable for practical implementation since in reality the bandwidth of the communication network and power capacity are usually constrained. The event-triggered algorithm overcomes some practical limitations, such as the unbounded growth of the adaptive gain and requirement of additional internal dynamics, by constructing a new triggering strategy. In addition, a continuous nonlinear function is used to approximate the signum function to reduce the chattering phenomenon in reality.

In distributed optimization of networked systems, each member has a local cost function, and the goal is to cooperatively minimize the sum of all the local cost functions. The distributed time-varying optimization problem is investigated for networked Lagrangian systems with parametric uncertainties in the dissertation. Usually, in the literature, to ad-

dress some distributed control problems for nonlinear systems, a networked virtual system is constructed, and a tracking algorithm is designed such that the agents' physical states track the virtual states. It is worth pointing out that such an idea requires the exchange of the virtual states and hence necessitates communication among the group. In addition, due to the complexities of the Lagrangian dynamics and the distributed time-varying optimization problem, there exist significant challenges. This dissertation proposes distributed time-varying optimization algorithms that achieve zero optimum-tracking errors for the networked Lagrangian agents without the communication requirement. The main idea behind the proposed algorithms is to construct a reference system for each agent to generate a reference velocity using absolute and relative physical state measurements with no exchange of virtual states needed, and to design adaptive controllers for Lagrangian systems such that the physical states are able to track the reference velocities and hence the optimal trajectory. The algorithms introduce mutual feedback between the reference systems and the local controllers via physical states/measurements and are amenable to implementation via local onboard sensing in a communication unfriendly environment. Specifically, first, a base algorithm is proposed to solve the distributed time-varying optimization problem for networked Lagrangian systems under fixed graph. Then, based on the base algorithm, a continuous function is introduced to approximate the signum function, forming a continuous distributed optimization algorithm and hence removing the chattering. Then, by using the structure of the base algorithm, a distributed time-varying optimization algorithm is designed for networked Lagrangian systems under switching graphs.

Contents

List of Figures

Chapter 1

Introduction

Due to the advantages in achieving group performance with low operation cost and flexible scalability, and potential practical applications in vehicle formation, sensor networks, cooperative surveillance, and so on [88, 95], distributed cooperative control of a group of robots/agents have drawn massive attention from various scientific communities. Consensus is an important research subject in distributed cooperative control of multi-agent systems, where all the agents reach an agreement on a state of interest. A number of distributed consensus algorithms have been proposed to solve the consensus problems for a group of agents with no leader [90, 94] and one leader [55, 10]. In this dissertation, we address three more complex and challenging problems for multi-agent systems, which are containment control, distributed average tracking and distributed time-varying optimization of multi-agent systems. We provide communication-reduced solutions to these problems:

1. Sampled-data containment control for double-integrator agents with dynamic leaders with nonzero inputs,

2. Robust distributed average tracking for double-integrator agents without velocity measurements under event-triggered communication,

3. Distributed time-varying optimization of networked Lagrangian systems.

Thus, we continue with an overview of containment control, distributed average tracking and distributed time-varying optimization problems.

1.1 Containment Control

Consider a collection of autonomous, mobile agents consisting of multiple leaders and followers, and the objective of the containment control [60] is to drive the followers to converge to the convex hull spanned by the leaders. Several natural phenomena exhibit the relationship between leaders and followers in the containment control problem. For instance, several sheepdogs gather a flock of sheep and guide them safely to a desired location [110]. Another biological example is provided in [53, 23], where female silkworm moths are capable of releasing a certain kind of pheromone to attract male moths to swarm in tight geometrical configurations. On the other hand, the containment control problem has practical applications. For instance, several robots capable of self-navigation are able to guide a group of agents to cross a partially known area [23]. Also, the containment control problem has applications in coordination of a group of robots [11].

A number of algorithms have been reported in the literature to deal with the containment control problem under various scenarios. For instance, containment control algorithms are proposed for a group of single-integrator agents [9, 72], double-integrator agents [11, 68, 72], and agents with general linear dynamics [70] and Euler-Lagrange dynamics [78].

The aforementioned results are derived for continuous-time cases, which require continuous sensing and interaction among agents. However, when each agent has limited power supply and sensing capacities, energy saving becomes one of the main factors that the designers have to take into account. Because of the advantages in cost reduction, the event-triggered and discrete-time containment control algorithms are studied in the literature.

Several different event-triggered containment control algorithms are proposed in the literature. See [131, 80, 24, 75, 22, 126, 74] for instance. These event-triggered containment control algorithms require that each agent continuously monitor the communication channels and certain states, and continuously compute and check the event-triggering functions to see whether they exceed some threshold. These actions will cost additional energy and resources. It is also worth noting that in [131, 80, 24, 75, 22, 126, 74], the leaders' inputs are either zero or designed to drive the leaders to some stationary locations, which are simpler than the case where the dynamic leaders' inputs can be arbitrary as long as they are bounded.

Discrete-time containment control algorithms are proposed for multi-agent systems with single-integrator dynamics [23, 112], double-integrator dynamics [2, 118, 73, 72], higher-order integrator dynamics [116, 98], and general discrete-time linear dynamics [70]. The containment control problem for heterogeneous multi-agent systems is addressed in [99], where the followers have single- and double-integrator dynamics and leaders are single-integrator agents.

Among these discrete-time containment control algorithms, the sampled-data based ones stand out because of their advantages in performance/accuracy, price and generality.

Also it is more coincident with practical applications in real life. For instance, sampled-data based algorithms are proposed in [73] and [72] to solve the containment control problem for multiple agents with fractional-order double-integrator dynamics and ordinary double-integrator dynamics, respectively.

In the above mentioned discrete-time containment control algorithms, however, the leaders' inputs remain zero, which greatly simplifies analysis and design. One natural question arises is how to solve the containment control problem for the case where leaders' inputs are nonzero. In this case, discontinuous algorithms are usually used to achieve containment control for continuous-time single- and double-integrator agents [11, 9]. However, the discontinuous algorithms proposed in [11, 9] require each agent to continuously interact with its neighbors, and it is not clear whether it is applicable for double-integrator agents in a sampled-data setting. A solution to the question is provided in [23] for discrete-time higher-order-integrator agents if the leaders' trajectories are described by polynomial functions. Such trajectories can be generated by integrator agents with polynomial inputs. However, it is not directly applicable when the followers' dynamics become complicated and the leaders' inputs are non-polynomial as considered in this thesis. Also, note that to implement the discrete-time containment control algorithm in [23], each double-integrator follower needs to store a great amount of historical state information to update its controller.

In the sampled-data setting, there exist new challenges for the containment control of double-integrator agents with dynamic leaders with nonzero inputs. The coexistence of the sampled-data setting, double-integrator dynamics and dynamic leaders with nonzero inputs, makes the containment control problem more difficult and complicated, and renders

the existing related results in the literature inapplicable. Therefore, the development of new sampled-data containment control algorithm is needed for double-integrator agents with dynamic leaders with nonzero inputs.

1.2 Distributed Average Tracking

During the recent decade, the distributed average tracking problem, which includes consensus and distributed tracking as special cases, is formulated and addressed in the literature. In the distributed average tracking problem, each agent has a time-varying reference signal, and the goal is to design controllers for the agents based on local information such that all the agents are able to track the average of these reference signals. Because of the time-varying tracking objective and the lack of access to error signals, the distributed average tracking problem is theoretically more challenging compared with consensus and distributed tracking problems.

In the literature, there are cases where each agent aims to only estimate the average of these reference signals, which is often termed as dynamic average consensus. Some applications, such as feature-based map merging [1], and distributed Kalman filtering [4], have been reported in the literature. Several linear distributed algorithms are established to deal with the dynamic average consensus problem for certain types of reference signals. For instance, the dynamic average consensus problem is solved in [104], [3] and [122] for reference signals with steady state values, with a common denominator in their Laplace transforms, and slowly varying reference signals, respectively. The dynamic average consensus problem is solved with bounded steady state error for a strongly connected, weight-balanced

interaction topology in [65], where the discrete-time counterparts are addressed as well. A class of nonlinear algorithms is proposed in [85] for reference signals with bounded deviations, and the dynamic average consensus error is bounded. A non-smooth algorithm is proposed in [14], which enables each agent to keep track of the average of a class of reference signals with bounded derivatives. More recently, combined with an adaptive scheme, two dynamic average consensus algorithms without correct initialization are proposed in [43] such that each agent is able to estimate the average of the reference signals. Also, a robust dynamic average consensus algorithm is proposed for directed networks, which guarantees an arbitrary prescribed small steady-state error bound.

The aforementioned algorithms focus on estimator design, and in reality, some tasks, such as region following formation control [13] and coordinated path planning [108], require that each agent has a certain dynamics, and the goal is to design controllers for each agent such that its physical states track the average of multiple time-varying reference signals. In this context, the term distributed average tracking is often used. A nonsmooth algorithm is presented in [21] for double-integrator agents. It requires that the accelerations of the individual reference signals be bounded. For general linear systems, the distributed average tracking problem is addressed in [128]. The distributed average tracking algorithms mentioned above need full state information (e.g., both positions and velocities for double-integrator agents) to update the controllers. However, in some practical applications, partial states may be unavailable due to technology and space limitations. Moreover, it is usually less accurate and more expensive to implement velocity measurements compared with position measurements. Hence, it is worth investigating the distributed average

tracking problem for double-integrator agents without using velocity measurements. In [50], the authors investigate the problem described above. However, in [50], the lower bounds of the design parameters depend on the bounds related to the reference signals and the graph information including the largest and smallest nonzero eigenvalues of the Laplacian matrix, which are global information and may be inaccessible to the agents. Also, the algorithm in [50] is sensitive to parameter selection as a certain parameter is required to be exactly equal to a certain value.

All these aforementioned continuous-time distributed average tracking algorithms require each agent to continuously interact with its neighbors. However, it may not be practical due to the constrained bandwidth of the communication network and power source. On the other hand, discrete-time distributed average tracking algorithms require agents to interact with each other periodically. It may result in a waste of network resources. Furthermore, with regard to general bounded reference signals, there usually exist tracking errors by using the discrete-time algorithms. Thus, it makes sense to employ event-triggered control strategies to address the distributed average tracking problem. They take advantage of opportunistic aperiodic sampling to improve efficiency. In [64], the authors extend the algorithm in [65] by incorporating an event-triggered communication strategy, but specific initialization is needed for a certain variable, and there exist non-zero tracking errors for general bounded reference signals. A robust dynamic average consensus algorithm under dynamic event-triggered communication is proposed in [45] for agents to estimate the average of the reference signals. These works focus on the estimation aspect of the distributed average tracking problem, where the agents' dynamics are essentially single integrators.

1.3 Distributed Time-varying Optimization

Recently, the distributed optimization problem has attracted a significant amount of attention from different research societies due to its wide applications in power systems, sensor networks, machine learning and so on. In distributed optimization of networked systems, each member has a local cost function, and the goal is to cooperatively minimize the sum of all the local cost functions. A number of distributed optimization algorithms have been presented in the literature. See [123] and the references therein for instance. These results (e.g., [123] and the references therein) usually assume fixed local cost functions for the agents. However, the local cost functions might be time-varying in many practical applications, which reflects the fact that the optimal point might be changing over time and forms an optimal trajectory. For example, in the economic dispatch problem [25], a group of power generators aim to meet the power demand and minimize the total generation cost, which is the summation of each generator's individual cost. In a day, the power demand of a specific region changes over time, and the cost to generate the same amount of power also changes due to the fluctuation of resource's price and availability. These two reasons would result in a time-varying cost function for each generator. Hence, it is meaningful to investigate the distributed time-varying optimization problem.

In the literature, there are extensive distributed discrete-time algorithms that solves the time-varying optimization problem. See [124, 71, 5, 102] for examples. There usually exist bounded convergence errors to the optimal trajectory by using the discrete-time algorithms. There is another body of literature on distributed continuous-time optimization algorithms for time-varying cost functions. These distributed continuous-time optimization

algorithms have various applications in practice. One important applications lies in the coordination of a team of robots, where each robot's dynamics are described by differential equations and the team objective is to track an optimal trajectory defined by all the team members' cost functions. For instance, by constructing a quadratic objective function for each agent, the distributed time-varying optimization algorithms can be applied to solve the distributed average tracking of multi-agent system (see Remark 34 later for details). A few distributed time-varying optimization algorithms are established for single-integrator agents [106, 25, 107], double-integrator agents [92] and agents with integrator-like nonlinear dynamics [57]. In reality, a broad class of robots can be modeled by Lagrangian dynamics, for example, the planar elbow manipulator and autonomous vehicles [105]. The Lagrangian dynamics, which are the focus of this part of the dissertaion, are more complicated than single and double integrators, and are different from and cannot be included as special cases by the model considered in [57]. The complexity of the dynamics creates more challenges to solve the distributed time-varying optimization problem.

Some results addressing distributed coordination problems (e.g., consensus, or more generally, distributed optimization) for agents with nonlinear dynamics introduce distributed observers or virtual systems at a higher level, where the agents communicate their observer states (virtual states independent of the agents' physical states/measurements) with neighbors such that the observer states or virtual states reach consensus on the desired optimal point/trajectory. Then control algorithms are designed for the agents to track the virtual states (serving as reference trajectories). However, due to the lack of physical states/feedback (e.g., agent positions) in the observers, the reference trajectories generated

by such an approach do not explicitly take into account the physical agents' interaction with the environment and their capability. Also, such an approach cannot be implemented based on local measurements via onboard sensors without communication in a communication unfriendly environment.

1.4 Contribution of Dissertation

In this dissertation, we focus on the following three problems:

1. Sampled-data containment control for double-integrator agents with dynamic leaders with nonzero inputs,

2. Robust distributed average tracking for double-integrator agents without velocity measurements under event-triggered communication,

3. Distributed time-varying optimization of networked Lagrangian systems.

The contributions of this dissertation are discussed as follows.

In the first part of the dissertation (e.g., Chapter 2), we address the containment control problem in a sampled-data setting for double-integrator agents with multiple dynamic leaders with nonzero inputs under directed communication networks. The contributions of this part are two-fold. First, a sampled-data based containment control algorithm is proposed for double-integrator agents, which eliminates the requirement of continuous sensing and interactions. It is more suitable for practical applications, since continuous sensing and interaction are not energy-efficient, and demand a larger portion of energy on board compared with periodic ones. Second, the proposed algorithm is proposed for the

case where there are multiple dynamic leaders with nonzero inputs, which is one of the main differences distinguishing our work from the existing discrete-time distributed containment control algorithms in the literature. By the proposed algorithm, we show that all the followers converge to the convex hull spanned by the leaders with bounded errors. Both the collective position and velocity containment control errors are bounded, and the ultimate bound of the overall containment control error is proportional to the sampling period.

In the second part of the dissertation (e.g., Chapter 3), we focus on an event-triggered mechanism to solve the distributed average tracking problem for double-integrator agents without using velocity measurements. Before deriving the event-triggered approach, we first present a base algorithm to solve the distributed average tracking under continuous communication. Then we present an event-triggered distributed average tracking algorithm that further removes the continuous communication requirement. In contrast, [50] considers the problem of distributed average tracking of double-integrator agents without using velocity measurements under continuous communication, which does not enjoy the benefit of the event-triggered algorithm proposed in this part. While the base algorithm has some connection with [50], it is worth mentioning that even this base algorithm has an advantage over [50] in the sense that no global information is needed for parameter design. We would also like to point out that the base algorithm has a different structure from the one in [50]. Such structure and its independence on global information lay a solid base for the development of the event-triggered algorithm. The proposed event-triggered algorithm is able to achieve distributed average tracking with zero tracking errors, does not require correct initialization, and is free of Zeno behavior. In contrast to [45], which is limited

to only single-integrator agents, double-integrator agents without using velocity measurements are considered in this part, which is a more complicated and challenging problem. It is also noted that there are some practical limitations for the algorithm in [45]. First, the time-varying gain may grow unbounded due to persistent disturbance, which would affect the convergence and the success of the event triggering scheme. Second, an extra internal dynamics is needed to ensure the exclusion of Zeno behavior, which may cost extra computational power and storage space. In addition, the use of the signum function may cause the chattering phenomenon in real applications. The proposed event-triggered algorithm overcomes the aforementioned limitations in [45]. In this algorithm, a new adaptive law and a new event-triggering strategy are constructed and a continuous nonlinear function is used to approximate the signum function.

In the third part of the dissertation (e.g., Chapter 4), we propose communication-free distributed time-varying optimization algorithms for networked Lagrangian agents with parametric uncertainties. The main idea of the proposed algorithms is constructing a reference system for each agent, which is driven by the physical states instead of virtual states between neighbors and generates a reference velocity, and then designing adaptive controllers such that the agents' physical states track their reference velocities, and hence the optimal trajectory. The algorithms introduce mutual influence/feedback between reference systems and local controllers via physical states/measurements and are amenable to implementation via local onboard sensing in a communication unfriendly environment. Due to the coupling and mutual influence of the reference systems and the agents' dynamics, there are significant new challenges in the convergence analysis. In particular, the reference

systems are rewritten as coupled and perturbed networked second-order systems by taking the tracking errors between agents' velocities and their own reference states as disturbances. Due to the use of the nonlinear functions (the signum function and the one in (4.17) later) in the construction of the reference systems, the coupled and perturbed networked systems have disturbances inside and outside the nonlinear functions, and the general input-to-state stability analysis might not be directly applicable. This requires novel rigorous analysis on the impact of disturbance on the optimum-tracking performance of the perturbed systems. To this end, this dissertation carefully examines the perturbed systems, and obtains the input-to-state-like stability from the disturbances to optimum-tracking errors. That is, the optimum-tracking errors remain bounded if the disturbances are bounded in a certain sense and converges to zero if the disturbances converge to zero (See Proposition 36 for instance). These intermediate results facilitate the convergence analysis of the proposed algorithm for the networked Lagrangian agents.

To be exact, we first design a base algorithm for the networked Lagrangian systems to achieve exact optimum tracking under fixed graph. Since the base algorithm uses the signum function to construct the reference systems, which might cause chattering during the implementation in practice. Built on the base algorithm, we then propose to approximate and replace the signum function in the reference systems with a smooth nonlinear function, generating continuous control torques for the Lagrangian systems. Such an approximation method can be found in [83, 34], where traditional stabilization of a single agent is addressed. While addressing the time-varying optimization of multi-agent systems is more complicated and the theoretical proof can not be directly implied from [83, 34]. Using a similar structure

of the base algorithm, we then design a distributed time-varying optimization algorithm for networked Lagrangian systems under switching graphs, which is more applicable than the base algorithm in the terms of the types of interaction graph among the agents. However, fixed Hessian matrices are assumed for the cost functions, and it is more restrictive compared with the base algorithm. It is also worth mentioning that the reference systems designed for the fixed and switching graph cases are different.

Comparison with Related Works. The works [127, 133, 132] focus on solving the distributed time-invariant optimization problem for networked Lagrangian agents. They follow the aforementioned distributed observer idea, which rely on the exchange of virtual states between neighbors. The work [127] also considers the case of time-invariant cost functions with additive uncertainties modeled by time-dependent functions, and nonzero bounded optimum-tracking errors are achieved. In contrast, the proposed algorithms in this part of the dissertation solves the optimization problem with time-varying cost functions, which is not addressed in [133, 132]. Compared with [127], the problem considered in this part is more general and can be solved with zero optimum-tracking errors. More importantly, the proposed algorithms relies purely on physical states without the need for exchange of virtual states and can be implemented in a communication unfriendly application. In contrast, the communication of virtual states between neighbors is necessary in [133, 132, 127]. The structure of the proposed algorithms is inspired by [115], where the consensus and leader-following tracking of networked Lagrangian systems are addressed. However, the problem considered in this part is more complex and challenging, and includes the consensus and leader-following tracking of networked agents as special cases. In

[92], the distributed time-varying optimization problem is solved for networked single and double integrators, and the method of signum function approximation is also applied to deal with the chattering issue while implemention. However, we consider the distributed time-varying optimization of networked Lagrangian systems, and the agents' dynamics are more complex. As an intermediate step in the convergence analysis, a perturbed networked second-order system is investigated. Compared with [92], where, essentially, a disturbance-free networked second-order systems is considered, additional analysis is needed to address the influence on the system performance for the coexistence of the disturbances and the signum function or its approximation.

1.5 Preliminaries

In the reminder of this chapter, we introduce notations, algebraic graph theory and Lagrange dynamics.

1.5.1 Notations

Throughout this thesis, let $\mathbb{R}$, $\mathbb{R}_{\geq 0}$ and $\mathbb{R}_+$ denote the sets of all real numbers, all nonnegative real numbers and all positive real numbers, respectively. For a set $\mathcal{S}$, $|\mathcal{S}|$ denotes the cardinality of $\mathcal{S}$, and for a real number $x \in \mathbb{R}$, $|x|$ denotes the absolute value of x. The transpose of matrix A is denoted by A^T. For a given vector $x = [x_1, \ldots, x_p]^T \in \mathbb{R}^p$, define $\|x\|_1 = \sum_{i=1}^p |x_i|$, $\|x\|_2 = \sqrt{|x_1|^2 + \cdots + |x_p|^2}$, and $\|x\|_\infty = \max_{i=1,\ldots,p} |x_i|$. For a symmetric matrix $A \in \mathbb{R}^{p \times p}$, let $\lambda_1(A) \leq \cdots \leq \lambda_p(A)$ denote its eigenvalues. The Kronecker product of matrices A and B is denoted by $A \otimes B$. Let $\text{diag}\{A_1, \ldots, A_p\}$,

where $A_i \in \mathbb{R}^{n \times m}$, represent the block diagonal matrix with the i-th block in the main diagonal being A_i. For a vector $x \in \mathbb{R}^p$, define $\text{sgn}(x) = [\text{sgn}(x_1), \ldots, \text{sgn}(x_p)]^T$ where $\text{sgn}(x_i) = 1$ if $x_i > 0$, $\text{sgn}(x_i) = 0$ if $x_i = 0$, and $\text{sgn}(x_i) = -1$ if $x_i < 0$. Let $\mathbf{0}_{m \times n} \in \mathbb{R}^{m \times n}$ and $\mathbf{1}_{m \times n} \in \mathbb{R}^{m \times n}$ denote the $m \times n$ dimensional zero and all-ones matrix, respectively, and for simplicity, let $\mathbf{0}_m = \mathbf{0}_{m \times 1}$ and let $\mathbf{1}_m = \mathbf{1}_{m \times 1}$. $I_n \in \mathbb{R}^{n \times n}$ denotes the identity matrix. For a time-varying function $f : \mathbb{R}^p \times \mathbb{R}_{\geq 0} \to \mathbb{R}$, its gradient, denoted by $\nabla f(q, t) \in \mathbb{R}^p$ with $q \in \mathbb{R}^p$ and $t \in \mathbb{R}_{\geq 0}$, is the partial derivative of $f(q, t)$ with respect to q, and its Hessian, denoted by $H(q, t) \in \mathbb{R}^{p \times p}$, is the partial derivative of the gradient $\nabla f(q, t)$ with respect to q. Define $\mathcal{L}_\infty^p = \left\{ x : [0, \infty) \to \mathbb{R}^p \mid \sup_{t \geq 0} \|x(t)\|_\infty < \infty \right\}$ and $\mathcal{L}_2^p = \left\{ x : [0, \infty) \to \mathbb{R}^p \mid \sqrt{\int_0^\infty x^T(t) x(t) \mathrm{d}t} < \infty \right\}$. A continuous function $\varpi : [0, a) \to [0, \infty)$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\varpi(0) = 0$. It is said to belong to class $\mathcal{K}_\infty$ if $a = \infty$ and $\varpi(r) \to \infty$ as $r \to \infty$. A continuous function $\rho : [0, a) \times [0, \infty) \to [0, \infty)$ is said to belong to class $\mathcal{KL}$ if, for each fixed s, the mapping $\rho(r, s)$ belongs to class $\mathcal{K}$ with respect to r and, for each fixed r, the mapping $\rho(r, s)$ is decreasing with respect to s and $\rho(r, s) \to 0$ as $s \to \infty$.

1.5.2 Graph Theory

For a multi-agent system consisting of N agents, the interaction topology can be modeled by an undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{1, \ldots, N\}$ and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denote the node set and edge set, respectively. An edge denoted by $(i, j) \in \mathcal{E}$, means that agent i and j can obtain information from each other. In an undirected graph, the edges (i, j) and (j, i) are equivalent. It is assumed that $(i, i) \notin \mathcal{E}$. The neighbor set of node i is denoted by $\mathcal{N}_i = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\}$. The adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{N \times N}$ of the graph G is defined

such that $a_{ij} = 1$ if $(j, i) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. For an undirected graph, $a_{ij} = a_{ji}$. The Laplacian matrix $L = [L_{ij}] \in \mathbb{R}^{N \times N}$ associated with the adjacency matrix A is defined as $L_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}$ and for $i \neq j$, $L_{ij} = -a_{ij}$. By arbitrarily assigning an orientation for every edge in $\mathcal{G}$, let $B = [B_{ij}] \in \mathbb{R}^{N \times |\mathcal{E}|}$ denote the incidence matrix associated with graph $\mathcal{G}$, where $B_{ij} = -1$ if edge e_j leaves node i, $B_{ij} = 1$ if it enters node i, and $B_{ij} = 0$ otherwise.

An undirected path between node i_1 and i_k is a sequence of edges of the form (i_1, i_2), (i_2, i_3), ..., (i_{k-1}, i_k), where $i_k \in \mathcal{V}$. A connected graph means that there exists an undirected path between any pair of nodes in $\mathcal{V}$.

1.5.3 Lagrange Dynamics

We consider N Lagrangian systems, and the interaction topology among these agents is characterized as the graph $\mathcal{G}$. The equations of motion of the i-th Lagrangian system can be described by [105]

$$M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + g_i(q_i) = \tau_i \tag{1.1}$$

where $q_i \in \mathbb{R}^p$ is the generalized position (or configuration), $M_i(q_i) \in \mathbb{R}^{p \times p}$ is the inertia matrix, $C_i(q_i, \dot{q}_i) \in \mathbb{R}^{p \times p}$ is the Coriolis and centrifugal matrix, $g_i(q_i) \in \mathbb{R}^p$ is the gravitational torque, and $\tau_i \in \mathbb{R}^p$ is the exerted control torque. Three well-known properties associated with the dynamics (1.1) are listed as follows [105, 46].

Property 1 *The inertial matrix $M_i(q_i)$ is symmetric and uniformly positive definite, and there exist positive constants $k_{\bar{C}}$ and $k_{\bar{g}}$ such that $\|C_i(q_i, \dot{q}_i)\|_2 \leq k_{\bar{C}} \|\dot{q}_i\|_2$ and $\|g_i(q_i)\|_2 \leq k_{\bar{g}}$, $\forall i \in \mathcal{V}$.*

Property 2 *The Coriolis and centrifugal matrix $C_i(q_i, \dot{q}_i)$ can be suitably chosen such that the matrix $\dot{M}_i(q_i) - 2C_i(q_i, \dot{q}_i)$ is skew-symmetric.*

Property 3 *The dynamics (1.1) depend linearly on an unknown constant parameter vector $\vartheta_i \in \mathbb{R}^m$, that is, for any $x, y \in \mathbb{R}^p$, it holds that*

$$M_i(q_i)x + C_i(q_i, \dot{q}_i)y + g_i(q_i) = Y_i(q_i, \dot{q}_i, y, x)\vartheta_i, \tag{1.2}$$

where $Y_i(q_i, \dot{q}_i, y, x)$ is the regressor matrix.

Chapter 2

Sampled-data Containment Control for Double-Integrator Agents with Dynamic Leaders with Nonzero Inputs

2.1 Problem Statement

Consider a network of n agents whose interactions are represented by the directed graph $\mathcal{G}$. Each agent i has double-integrator dynamics given by

$$\dot{r}_i(t) = v_i(t), \quad \dot{v}_i(t) = u_i(t), \quad i = 1, \ldots, n,$$

where $r_i(t) \in \mathbb{R}^p$ and $v_i(t) \in \mathbb{R}^p$ denote the position and velocity of agent i at time t, respectively, and $u_i(t)$ is the corresponding control input. In this part, we consider a

19

sampled-data setting where the agents have continuous-time dynamics while the control inputs are based on zero-order hold and the interactions with neighbors are made at discrete sampling times. Then the system can be discretized as

$$r_i[k+1] = r_i[k] + Tv_i[k] + \frac{T^2}{2}u_i[k]$$

$$v_i[k+1] = v_i[k] + Tu_i[k],$$

(2.1)

where T is the sampling period, k is the discrete-time index, and $r_i[k] \in \mathbb{R}^p$, $v_i[k] \in \mathbb{R}^p$ and $u_i[k] \in \mathbb{R}^p$ represent the position, velocity, and control input of the ith agent at $t = kT$, respectively.

We adopt the definitions of the leaders and the followers used in [8]. That is, an agent is called a leader if and only if it has no neighbor, and otherwise it is called a follower. Without loss of generality, let $\mathscr{F} = \{1, \ldots, m\}$ and $\mathscr{L} = \{m+1, \ldots, n\}$ denote the follower set and the leader set, respectively. Therefore, the row-stochastic matrix D associated with the directed graph $\mathcal{G}$ can be written as

$$D = \begin{bmatrix} D_1 & D_2 \\ \mathbf{0}_{(n-m)\times m} & I_{(n-m)\times(n-m)} \end{bmatrix}$$

where $D_1 \in \mathbb{R}^{m\times m}$ and $D_2 \in \mathbb{R}^{m\times(n-m)}$. We assume that $\mathcal{G}$ satisfies the following assumption.

Assumption 1 *For each of the followers, there is at least one leader that has a directed path to the follower.*

Lemma 1 *[70] Under Assumption 1, the matrix D_1 has all the eigenvalues within the unit circle, and each entry of $(I_m - D_1)^{-1}D_2$ is nonnegative, and each row of $(I_m - D_1)^{-1}D_2$ has a sum equal to 1.* □

Definition 2 *Let $\mathcal{C}$ be a set in a real vector space $\mathcal{S} \subseteq \mathbb{R}^p$. The set is convex if, for any x and y in $\mathcal{C}$, the point $(1-\alpha)x + \alpha y \in \mathcal{C}$ for any $\alpha \in [0,1]$. The convex hull for a set of points $\mathcal{X} := \{x_1, \ldots, x_m\}$ in $\mathcal{S}$, denoted by $\mathrm{Co}(\mathcal{X})$, is the minimal convex set containing all points in $\mathcal{X}$, that is, $\mathrm{Co}(\mathcal{X}) := \{\sum_{i=1}^m \beta_i x_i \mid x_i \in \mathcal{X}, \beta_i \geq 0, \sum_{i=1}^m \beta_i = 1\}$.*

In this chapter, the objective is to solve the containment control problem, that is to design $u_i[k]$ for follower i, $i \in \mathscr{F}$, by using its own and neighbors' states, $\{r_j\}_{j \in \mathcal{N}_i \cup \{i\}}$ and $\{v_j\}_{j \in \mathcal{N}_i \cup \{i\}}$, such that all followers' positions and velocities converge to the convex hull spanned by the dynamic leaders' positions and velocities, respectively, which are given by $\mathrm{Co}(\{r_l\}_{l \in \mathscr{L}})$ and $\mathrm{Co}(\{v_l\}_{l \in \mathscr{L}})$, respectively.

Note that by properly designing $u_i[k]$ for leader $i \in \mathscr{L}$, the leader will be able to follow a certain desired trajectory. Then, the leaders are capable of guiding the followers through a certain region safely and reach the desired location. We assume that the leaders' inputs are pre-designed and satisfy the following condition.

Assumption 2 *The inputs of the leaders are bounded, i.e., for any $j \in \mathscr{L}$, $\|u_j[k]\|_2 \leq c_1$, where c_1 is a positive constant.*

2.2   Sampled-data Containment Control

In order to solve the multi-agent containment control problem, we consider the following controller for follower i as

$$u_i[k] = \sum_{j \in \mathscr{L} \cup \mathscr{F}} d_{ij} \left(\frac{v_j[k] - v_j[k-1]}{T} - \gamma_1 \{r_i[k] - r_j[k]\} - \gamma_2 \{v_i[k] - v_j[k]\} \right), \quad i \in \mathscr{F}$$

$$(2.2)$$

where d_{ij} is the $(i,j)$the entry of the matrix D, and $\gamma_1, \gamma_2 > 0$ are constant. Essentially, the term $\frac{v_j[k] - v_j[k-1]}{T}$ make use of past data to approximate the acceleration of agent j. Therefore, each follower only uses its own and its neighbors's current and previous velocities as well as the current positions to update its control input, which means the algorithm (3.5) can be implemented in reality.

Define the position and velocity containment control errors for follower i as $x_i[k] = \sum_{j=1}^n d_{ij}(r_i[k] - r_j[k])$ and $y_i[k] = \sum_{j=1}^n d_{ij}(v_i[k] - v_j[k])$, respectively. Define the collective position and velocity containment control error as $X[k] = \left(x_1^\top[k], \ldots, x_m^\top[k]\right)^\top$ and $Y[k] = \left(y_1^\top[k], \ldots, y_m^\top[k]\right)^\top$, respectively. Using (3.5) for (2.1), we have

$$X[k+1] = \left(A_{11} \otimes I_p\right)X[k] + \left(A_{12} \otimes I_p\right)Y[k] - \left(\frac{T}{2}D_1 \otimes I_p\right)Y[k-1] + \left(\frac{T}{2}D_2 \otimes I_p\right)\Delta[k]$$

$$Y[k+1] = \left(A_{21} \otimes I_p\right)X[k] + \left(A_{22} \otimes I_p\right)Y[k] - \left(D_1 \otimes I_p\right)Y[k-1] + \left(D_2 \otimes I_p\right)\Delta[k],$$

where

$$A_{11} = \left(1 - \frac{T^2}{2}\gamma_1\right)I_m + \frac{T^2}{2}\gamma_1 D_1,$$

$$A_{12} = \left(T - \frac{T^2}{2}\gamma_2\right)I_m + \left(\frac{T}{2} + \frac{T^2}{2}\gamma_2\right)D_1,$$

$$A_{21} = -T\gamma_1\left(I_m - D_1\right),$$

$$A_{22} = \left(1 - T\gamma_2\right)I_m + \left(1 + T\gamma_2\right)D_1.$$

and $\Delta[k] = 2v_L[k] - v_L[k+1] - v_L[k-1]$ with $v_L[k] = \left(v_{m+1}^\top[k], \ldots, v_n^\top[k]\right)^\top$. Define $Z[k+1] = \left(X^\top[k+1], Y^\top[k+1], X^\top[k], Y^\top[k]\right)^\top$. It then follows that

$$Z[k+1] = \widetilde{A}Z[k] + \widetilde{B}\Delta[k], \tag{2.3}$$

where $\widetilde{A} = A \otimes I_p$ with

$$
A = \begin{bmatrix}
A_{11} & A_{12} & \mathbf{0}_{m \times m} & -\frac{T}{2} D_1 \\
A_{21} & A_{22} & \mathbf{0}_{m \times m} & -D_1 \\
I_m & \mathbf{0}_{m \times m} & \mathbf{0}_{m \times m} & \mathbf{0}_{m \times m} \\
\mathbf{0}_{m \times m} & I_m & \mathbf{0}_{m \times m} & \mathbf{0}_{m \times m}
\end{bmatrix},
\tag{2.4}
$$

and $\widetilde{B} = \left[\frac{T}{2}, 1, 0, 0 \right]^\top \otimes \left(D_2 \otimes I_p \right)$.

The eigenvalues of $\widetilde{A}$ play an important role in determining the solution of (2.3). Therefore, we investigate the eigenvalues of $\widetilde{A}$ in the following. We first present three useful lemmas before moving on.

Lemma 3 *(Generalized Schur's Formula [66]) Let $M_{ij} \in \mathbb{R}^{n \times n}$, $i, j \in \mathcal{M}$, where $\mathcal{M} = \{1, \dots, m\}$, and*

$$
M = \begin{bmatrix}
M_{11} & \cdots & M_{1m} \\
\vdots & \ddots & \vdots \\
M_{m1} & \cdots & M_{mm}
\end{bmatrix}.
$$

If M_{ij}, $i, j \in \mathcal{M}$ pairwise commute, i.e., $M_{ij} M_{ls} = M_{ls} M_{ij}$ for all possible pairs of indices i, j and l, s, then

$$
\det(M) = \det \left(\sum_{\pi \in \mathcal{S}_m} \mathrm{sgn}(\pi) M_{1\pi(1)} M_{2\pi(2)} \dots M_{m\pi(m)} \right),
$$

where $\det(\cdot)$ denotes the determinant of a matrix, π is a permutation, set $\mathcal{S}_m$ denotes the set of all possible permutations of the $\mathcal{M}$, and $\mathrm{sgn}(\pi)$ denotes the parity of the permutation π. $\qquad \square$

Lemma 4 *Let $P(z)$ be a polynomial of order three with complex coefficients in the form of $P(z) = z^3 + \alpha_1 z^2 + \alpha_2 z + \alpha_3$, where $\alpha_i = p_i + \boldsymbol{j} q_i$, $i = 1, \dots, 3$ and $\boldsymbol{j}$ is the imaginary unit.*

The polynomial $P(z)$ has all its zeros in the open left half of the z-complex plane if and only if $p_1 > 0$, $p_1^2 p_2 + p_1 q_1 q_2 - p_1 p_3 - q_2^2 > 0$, and $\det(M_3) > 0$ where

$$
M_3 = \begin{bmatrix}
p_1 & p_3 & p_5 & -q_2 & -q_4 \\
1 & p_2 & p_4 & -q_1 & -q_3 \\
0 & p_1 & p_3 & 0 & -q_2 \\
0 & q_2 & q_4 & p_1 & p_3 \\
0 & q_1 & q_3 & p1 & p_2
\end{bmatrix}.
$$

Proof: This lemma is a special case of the Theorem 3.2 in [39], and the proof is thus omitted. $\qquad\square$

Lemma 5 *[56] The matrix $M \in \mathbb{R}^{n \times n}$ has eigenvalues λ_1, ..., λ_n. Let $g(x) = a_0 + a_1 x + \cdots + a_k x^k$ be a polynomial, and let $g(M) = a_0 I_n + a_1 M + \cdots + a_k M^k$. Then the eigenvalues of $g(M)$ are $g(\lambda_1)$, ..., $g(\lambda_n)$.* $\qquad\square$

With the above three lemmas, we can obtain the following results on the eigenvalues of the matrix $\widetilde{A}$.

Lemma 6 *Suppose that Assumption 1 holds. Let λ_i be the ith eigenvalue of D_1. The matrix $\widetilde{A}$ has all eigenvalues within the unit circle if and only if there exist positive scalars T, γ_1 and γ_2 such that*

$$
\frac{2\gamma_2}{T\gamma_1} > \frac{1 - |\lambda_i|^2}{|1 - \lambda_i|^2}, \quad i = 1, \ldots, m, \tag{2.5}
$$

and

$$
\left(\frac{2\gamma_2}{T} - \frac{1 - |\lambda_i|^2}{|1 - \lambda_i|^2} \gamma_1 \right)^2 \left[\frac{2(1 - |\lambda_i|^2)}{|1 - \lambda_i|^2} - T\gamma_2 \right] - \frac{16(\mathrm{Im}\,\{\lambda_i\})^2 \gamma_2^3}{|1 - \lambda_i|^4 T} > 0, \quad i = 1, \ldots, m, \tag{2.6}
$$

hold. In addition, such positive scalars T, γ_1 and γ_2 always exist.

24

Proof: First, we prove that the matrix A defined in (2.4) has all eigenvalues within the unit circle if and only if there exist positive scalars T, γ_1 and γ_2 such that (2.5) and (2.6) hold. Note that the characteristic polynomial of A is given by

$$
\det(sI_{4m} - A) = \det \left(\begin{bmatrix} sI_m - A_{11} & -A_{12} & \mathbf{0}_{m \times m} & \frac{T}{2}D_1 \\[2mm] -A_{21} & sI_m - A_{22} & \mathbf{0}_{m \times m} & D_1 \\[2mm] -I_m & \mathbf{0}_{m \times m} & sI_m & \mathbf{0}_{m \times m} \\[2mm] \mathbf{0}_{m \times m} & -I_m & \mathbf{0}_{m \times m} & sI_m \end{bmatrix} \right)
$$

$$
= \det \left(s \Big[s(sI_m - A_{11})(sI_m - A_{22}) - sA_{12}A_{21} \right.
$$

$$
+ (sI_m - A_{11})D_1 - \frac{1}{2}D_1 A_{21} \Big] \Big)
$$

$$
= \det \left(s \left\{ \Big[s^3 - \Big(2 - \frac{T^2}{2}\gamma_1 - T\gamma_2 \Big) s^2 + \Big(1 + \frac{T^2}{2}\gamma_1 \right.\right.
$$

$$
\left. - T\gamma_2 \Big) s \Big] I_m + \Big[\Big(-1 - \frac{T^2}{2}\gamma_1 - T\gamma_2 \Big) s^2 \right.
$$

$$
\left.\left. + \Big(2 - \frac{T^2}{2}\gamma_1 + T\gamma_2 \Big) s - 1 \Big] D_1 \right\} \right),
$$

where we have used Lemma 3 to obtain the second to the last equality because $sI_m - A_{11}$, $-A_{12}$, $\mathbf{0}_{m \times m}$, $\frac{T}{2}D_1$, $-A_{21}$, $sI_m - A_{22}$, D_1, $-I_m$ and sI_m commute pairwise. Let $\lambda_1, \ldots, \lambda_m$ be the eigenvalues of D_1. Then by Lemma 5 and the fact that the determinant of a matrix is the product of its eigenvalues, it holds that $\det[g_1(s)I_m + g_2(s)D_1] = \prod_{i=1}^{m}[g_1(s) + g_2(s)\lambda_i]$, where g_1 and g_2 are two polynomial functions of s. Thus, it follows that

$$
\det(sI_{4m} - A) = \prod_{i=1}^{m} \left(s \left\{ s^3 - \Big(2 - \frac{T^2}{2}\gamma_1 - T\gamma_2 \Big) s^2 + \Big(1 + \frac{T^2}{2}\gamma_1 \right.\right.
$$

$$
\left. - T\gamma_2 \Big) s + \Big[\Big(-1 - \frac{T^2}{2}\gamma_1 - T\gamma_2 \Big) s^2 \right.
$$

$$
\left.\left. + \Big(2 - \frac{T^2}{2}\gamma_1 + T\gamma_2 \Big) s - 1 \Big] \lambda_i \right\} \right).
$$

Thus, the roots of $\det(sI_{4m} - A) = 0$ either equal to zero or satisfy

$$
\begin{aligned}
s^3 &+ \left[-2 - \lambda_i + (1 - \lambda_i)\left(\frac{T^2}{2}\gamma_1 + T\gamma_2\right)\right]s^2 \\
&+ \left[1 + 2\lambda_i + (1 - \lambda_i)\left(\frac{T^2}{2}\gamma_1 - T\gamma_2\right)\right]s - \lambda_i = 0.
\end{aligned}
\tag{2.7}
$$

It is trivial when the roots of $\det(sI_{4m} - A) = 0$ are zero. Note that the matrix A has all

eigenvalues within the unit circle if and only if, for any eigenvalue of D_1, the roots of (2.7)

all lie inside the unit circle. Instead of computing the roots of (2.7) directly, we apply the

bilinear transformation $s = \frac{z+1}{z-1}$ to (2.7), which yields

$$
\begin{aligned}
(1 - \lambda_i)T^2\gamma_1 z^3 &+ 2(1 - \lambda_i)T\gamma_2 z^2 + (1 - \lambda_i)(4 - T^2\gamma_1)z \\
&+ 4(1 + \lambda_i) - 2(1 - \lambda_i)T\gamma_2 = 0.
\end{aligned}
\tag{2.8}
$$

Such bilinear transformation maps the left half of the complex z-plane to the interior of

the unit circle in the s-plane, it then follows that (2.7) has all the roots within the unit

circle if and only if (2.8) has all the roots in the open left half of the complex plane. Since

Assumption 1 holds, it follows that $|1 - \lambda_i| > 0$ by Lemma 1. Note that both γ_1 and the

sampling period T are positive. Then (2.8) is equivalent to

$$
z^3 + \alpha_1 z^2 + \alpha_2 z + p_3 + \boldsymbol{j}q_3 = 0.
\tag{2.9}
$$

where $\alpha_1 = \frac{2T\gamma_2}{T^2\gamma_1}$, $\alpha_2 = \frac{4 - T^2\gamma_1}{T^2\gamma_1}$, $p_3 = \frac{4(1 - |\lambda_i|^2)}{|1 - \lambda_i|^2 T^2\gamma_1} - \frac{2T\gamma_2}{T^2\gamma_1}$, and $q_3 = \frac{8\mathrm{Im}\{\lambda_i\}}{|1 - \lambda_i|^2 T^2\gamma_1}$. Denote by

$P(z, \lambda_i)$ the left hand side of (2.9) for some given λ_i. Note that for a given λ_i, $P(z, \lambda_i)$ is a

polynomial in the indeterminate z of degree 3. Then for a given λ_i, by Lemma 4, $P(z, \lambda_i)$

has all the zeros in the open left half of the complex plane if and only if the positive scalars

T, γ_1 and γ_2 satisfy

$$
f_j^i > 0, \quad j = 1, \ldots, 3, \quad i = 1, \ldots, m,
\tag{2.10}
$$

26

with $f_1^i = \alpha_1$, $f_2^i = \alpha_1\alpha_2 - p_3$, and $f_3^i = (\alpha_1\alpha_2 - p_3)^2 p_3 - q_3^2\alpha_1^3$. It is easy to see that

(2.5) follows from $f_1^i > 0$ and $f_2^i > 0$, $i = 1,\ldots,m$. Note that $f_3^i > 0$ can be written as

(2.6). Thus, the matrix A has all eigenvalues within the unit circle if and only if there exist

positive scalars T, γ_1 and γ_2 such that (2.5) and (2.6) hold.

By the fact that μ is an eigenvalue of $\widetilde{A}$ if and only if μ is also an eigenvalue of

A, we conclude that $\widetilde{A}$ has all eigenvalues within the unit circle if and only if there exist

positive scalars T, γ_1 and γ_2 such that (2.5) and (2.6) hold.

In the following, we show that such positive scalars T, γ_1 and γ_2 always exist.

Obviously, $f_1^i > 0$ $\forall i = 1,\ldots,m$. We rewrite f_2^i and f_3^i, $i = 1,\ldots,m$ as

$$f_2^i = \left(\frac{8\gamma_2}{\gamma_1^2}\right)\beta^3 - \left[\frac{4(1 - |\lambda_i|^2)}{|1 - \lambda_i|^2\gamma_1}\right]\beta^2,$$

$$f_3^i = 32\left\{\left[\frac{8(1 - |\lambda_i|^2)\gamma_2^2}{|1 - \lambda_i|^2\gamma_1^5}\right]\beta^8 - \left[\frac{8(1 - |\lambda_i|^2)^2\gamma_2}{|1 - \lambda_i|^4\gamma_1^4}\right.\right.$$
$$\left. + \frac{4\gamma_2^3}{\gamma_1^5} + \frac{16\left(\mathrm{Im}\left\{\lambda_i\right\}\right)^2\gamma_2^3}{|1 - \lambda_i|^4\gamma_1^5}\right]\beta^7 + \left[\frac{4(1 - |\lambda_i|^2)\gamma_2^2}{|1 - \lambda_i|^2\gamma_1^4}\right.$$
$$\left.\left. + \frac{2(1 - |\lambda_i|^2)^3}{|1 - \lambda_i|^6\gamma_1^3}\right]\beta^6 - \left[\frac{(1 - |\lambda_i|^2)^2\gamma_2}{|1 - \lambda_i|^4\gamma_1^3}\right]\beta^5\right\},$$

where $\beta = \frac{1}{T}$. Note that f_2^i and f_3^i are three polynomials in the indeterminate β of degree

3 and 8, respectively. The leading coefficients (the coefficient of the term with the highest

degree) of f_2^i and f_3^i are $e_2^i := \frac{8\gamma_2}{\gamma_1^2}$, and $e_3^i := \frac{256(1 - |\lambda_i|^2)\gamma_2^2}{|1 - \lambda_i|^2\gamma_1^5}$, respectively. We can see that

$e_2^i > 0$ $\forall i = 1,\ldots,m$, since $\gamma_1, \gamma_2 > 0$. When Assumption 1 holds, the eigenvalues of D_1 are

located inside the unit circle by Lemma 1, which implies that $1 - |\lambda_i|^2 > 0$ $\forall i = 1,\ldots,m$.

It then holds that $e_3^i > 0$ $\forall i = 1,\ldots,m$. Then it follows that, for any eigenvalue of D_1, and

any given positive constants γ_1 and γ_2,

$$\lim_{T \to 0^+} f_j^i = +\infty, \quad j = 2, 3.$$

27

Hence, given any eigenvalue of D_1, and for any positive finite constants γ_1 and γ_2, there always exists a positive constant $\overline{T}_{\lambda_i}$ such that for any $T < \overline{T}_{\lambda_i}$, $f_2^i > 0$ and $f_3^i > 0$ hold. Let $\overline{T} = \min_{i=1,\ldots,m} \overline{T}_{\lambda_i}$. When $T < \overline{T}$, f_j^i, $j = 1,\ldots,3$, hold for any eigenvalues of D_1, which implies the existence of these three positive scalars T, γ_1 and γ_2 such that (2.10) holds for any eigenvalue of D_1. $\qquad\square$

Theorem 7 *Let Assumptions 1 and 4 hold. If the positive scalars T, γ_1 and γ_2 satisfy (2.10) for any eigenvalue of D_1, using the algorithms (3.5) for (2.1), the followers converge to the convex hull spanned by the leaders with bounded position and velocity containment control error, and the overall containment control error, $\|X[k]\|_2 + \|Y[k]\|_2$, is ultimately bounded by $2c_1 c_2 T \sqrt{n-m}\left\|\widetilde{B}\right\|_2 / (1-\rho)$, where c_1 is given in Assumption 4, and positive constant c_2 and $\rho \in [0,1)$ satisfy $\left\|\widetilde{A}^j\right\|_2 \le c_2 \rho^j$, $j \ge 0$.*

Proof: It follows that the solution of (2.3) is

$$Z[k] = \widetilde{A}^k Z[0] + \sum_{i=0}^{k-1} \widetilde{A}^{k-i-1} \widetilde{B} \Delta[i].$$

Then, it holds that

$$\|Z[k]\|_2 \le \left\|\widetilde{A}^k Z[0]\right\|_2 + \left\|\sum_{i=0}^{k-1} \widetilde{A}^{k-i-1} \widetilde{B} \Delta[i]\right\|_2$$

$$\le \left\|\widetilde{A}^k\right\|_2 \|Z[0]\|_2 + 2\sqrt{n-m}Tc_1 \left\|\sum_{i=0}^{k-1} \widetilde{A}^{k-i-1}\right\|_2 \left\|\widetilde{B}\right\|_2,$$

where we have used the fact that

$$\|\Delta[i]\|_2 = \|2\boldsymbol{v}_L[i] - \boldsymbol{v}_L[i+1] - \boldsymbol{v}_L[i-1]\|_2$$

$$\le \|\boldsymbol{v}_L[i+1] - \boldsymbol{v}_L[i]\|_2 + \|\boldsymbol{v}_L[i] - \boldsymbol{v}_L[i-1]\|_2$$

$$\le 2\sqrt{n-m}Tc_1$$

holds for all i if Assumption 4 holds. Since Assumption 1 holds, and by Lemma 6, if the positive scalars T, γ_1, and γ_2 satisfy (2.10) for any eigenvalue of D_1, the matrix $\widetilde{A}$ has all the eigenvalues within the unit circle. Then by [67, 61], there exist two finite positive constants c_2 and $\rho \in [0, 1)$ such that $\left\|\widetilde{A}^j\right\|_2 \leq c_2 \rho^j$. Then, we have $\|Z[k]\|_2 \leq c_2 \rho^k \|Z[0]\|_2 + 2Tc_1c_2\sqrt{n-m}(1-\rho^k)\left\|\widetilde{B}\right\|_2/(1-\rho) < \infty$, which implies that both the position and velocity containment error are bounded. It also follows that $\lim_{k\to\infty} \|Z[k]\|_2 \leq 2Tc_1c_2\sqrt{n-m}\left\|\widetilde{B}\right\|_2/(1-\rho)$, since $\lim_{k\to\infty} \rho^k = 0$. Therefore, it holds that $\lim_{k\to\infty}(\|X[k]\|_2 + \|Y[k]\|_2) \leq \lim_{k\to\infty} \sqrt{2\|X[k]\|_2^2 + 2\|Y[k]\|_2^2} = \lim_{k\to\infty} \|Z[k]\|_2 \leq 2Tc_1c_2\sqrt{n-m}\left\|\widetilde{B}\right\|_2/(1-\rho)$. This completes the proof. $\qquad\square$

Remark 8 *The ultimate overall containment control error is proportional to the sampling period T. As $T \to 0$, $\|X[k]\|_2 + \|Y[k]\|_2 \to 0$, which implies that the position and velocity containment errors for each follower approach zero eventually.*

Remark 9 *The discrete-time controller (3.5) is robust to bounded state disturbance. Consider that $r_i[k+1] = r_i[k] + Tv_i[k] + \frac{T^2}{2}u_i[k] + d_i^r[k]$, $v_i[k+1] = v_i[k] + Tu_i[k] + d_i^v[k]$, where $d_i^r[k]$ and $d_i^v[k]$ are the position and velocity disturbances, respectively. Since (2.3) is a linear time-invariant system, then under the state disturbances, the followers are still capable of converging to the convex hull with bounded errors, the value of which depends on the bounded disturbances, in addition to the sampling period and the choices of positive scalars γ_1 and γ_2.*

The real-world communication environment may be corrupted by noise, that is, each agent has access to noisy state information received from its neighbors. In the present case, each agent i receives $\hat{r}_j[k] = r_j[k] + \boldsymbol{n}_j^r[k]$ and $\hat{v}_j[k] = v_j[k] + \boldsymbol{n}_j^v[k]$ from its neighbor

i, where $\boldsymbol{n}_j^r$ and $\boldsymbol{n}_j^v \in \mathbb{R}^p$ are noise vectors. The entries of each noise vector are drawn independently from some identical zero-mean distribution. Then by using noisy transmitted position and velocity information in the control law (3.5), that is, replacing $r_j[k]$ and $v_j[k]$ with $\hat{r}_j[k]$ and $\hat{v}_j[k]$, the followers are to converge to the convex hull spanned by the leaders with bounded error in expectation. The variance of the resulting overall containment control error is also bounded.

Remark 10 *The containment control problem for double-integrator agents in a sampled-data setting is also investigated in [72]. However, in [72], the leaders are with zero inputs, which can be included as a special case of this chapter. Moreover, when the leaders has nonzero inputs, the sampled-data containment algorithm proposed in [72] does not work anymore. It is also worth noting that the analysis and controller design have been greatly simplified under the assumption of zero inputs for the leaders.*

2.2.1 Selection of the Sampling Period

The sampling period T plays an essential role in ensuring that the matrix $\widetilde{A}$ has all its eigenvalues inside the unit circle and thus the convergence. Although it has been proven from Lemma 6, that given a communication network $\mathcal{G}$ satisfying Assumption 1 and some positive scalars γ_1 and γ_2, one can always find small enough sampling period T such that (2.5) and (2.6) hold, it is still not clear about how to choose appropriate sampling period T. We address this problem in the following.

Given any positive scalars γ_1 and γ_2, we can obtain the solution to (2.5) as

$$\mathcal{S}_1 = \left\{ T \mid 0 < T < \min_{i=1,\dots,m} \left\{ \frac{2|1-\lambda_i|^2 \gamma_2}{(1-|\lambda_i|^2)\gamma_1} \right\} \right\}. \tag{2.11}$$

30

The inequality (2.6) can be equivalently expressed as

$$a_3^i T^3 - a_2^i T^2 + a_1^i T - a_0^i < 0, \tag{2.12}$$

where $a_3^i = (1 - |\lambda_i|^2)^2 |1 - \lambda_i|^2 \gamma_1^2 \gamma_2$, $a_2^i = 2(1 - |\lambda_i|^2)\gamma_1 \left[(1 - |\lambda_i|^2)^2 \gamma_1 + 2|1 - \lambda_i|^4 \gamma_2^2 \right]$, $a_1^i = 4|1 - \lambda_i|^2 \gamma_2 \left\{ \left[|1 - \lambda_i|^4 + 4 \big(\mathrm{Im} \, \{\lambda_i\} \big)^2 \right] \gamma_2^2 + 2(1 - |\lambda_i|^2)^2 \gamma_1 \right\}$ and $a_0^i = 8(1 - |\lambda_i|^2)|1 - \lambda_i|^4 \gamma_2^2$.

The left-hand side of the inequality in (2.12), denoted by $g_i(T)$, is a polynomial of T with order 3. There are at most three roots for $g_i(T) = 0$, and then set $\mathcal{S}_2^i$ can be obtained. Let $\mathcal{S}_2 = \bigcap_{i=1}^m \mathcal{S}_2^i$. Therefore, we have the following corollary.

Proposition 11 *Suppose that Assumption 1 holds. Let λ_i be the ith eigenvalues of D_1. The matrix $\widetilde{A}$ has all eigenvalues within the unit circle if and only if the sampling period $T \in \mathcal{S}_1 \cap \mathcal{S}_2$.* $\square$

Note that from Lemma 6, a small enough sampling period T always exists such that the eigenvalues of $\widetilde{A}$ are located inside the unit circle. The following corollary gives a rough idea on how to choose a small enough sampling period given the underlying communication networks and positive scalars γ_1 and γ_2.

Corollary 12 *Suppose that Assumption 1 holds. Let λ_i be the ith eigenvalue of D_1. Given positive scalars γ_1 and γ_2, the matrix $\widetilde{A}$ has all eigenvalues within the unit circle if the sampling period $T \in (0, T_a)$ where*

$$T_a = \min_{i=1,\ldots,m} \left\{ \frac{2(1 - |\lambda_i|^2)|1 - \lambda_i|^2}{\left[|1 - \lambda_i|^4 + 4 \big(\mathrm{Im} \, \{\lambda_i\} \big)^2 \right] \gamma_2^2 + 2(1 - |\lambda_i|^2)^2 \gamma_1} \right\}.$$

Proof: The sets of $\mathcal{S}_1$ has been derived in (2.11). We focus on solve the inequality (2.12). Note that $a_3^i T^3 - a_2^i T^2 < 0$ if $0 < T < \frac{a_2^i}{a_3^i}$, and $a_1^i T - a_0^i < 0$ if $0 < T < \frac{a_0^i}{a_1^i}$. Then,

31

the inequality (2.12) holds if $0 < T < \min\left\{\frac{a_2^i}{a_3^i}, \frac{a_0^i}{a_1^i}\right\}$. In addition, it can be verified that

$\frac{a_0^i}{a_1^i} < \frac{2|1-\lambda_i|^2\gamma_2}{(1-|\lambda_i|^2)\gamma_1} < \frac{a_2^i}{a_3^i}$. Therefore, if $0 < T < \min_{i=1,\dots,m}\left\{\frac{a_0^i}{a_1^i}\right\} = T_a$, (2.5) and (2.6) hold. $\square$

Note that Lemma 6 and Proposition 11 give necessary and sufficient conditions such that $\widetilde{A}$ has all its eigenvalues inside the unit circle, and Corollary 12 only provides a conservative interval for the sampling period T, which is a sufficient condition.

It can be seen that the sampling period T should be small enough if the design parameter γ_1 and γ_2 are chosen to be large numbers. This observation coincides with the proof of Lemma 5. Also, note that as $T \to 0$, the controller (2) turns into a continuous-time controller, and it is well-known that the followers are to converge to the convex hull as long as γ_1 and γ_2 are positive. However, the resulting continuous-time controller as $T \to 0$ cannot be implemented in practice since each agent's input depends on its neighbors' inputs while the neighbors' inputs depend on their neighbors' inputs, which creates algebraic loops. In contrast, the introduced control algorithm (2) uses data from neighboring agents and can be implemented in a distributed manner in reality.

Given a graph satisfying Assumption 1, and any γ_1 and γ_2, to implement control algorithm (3.5) in practice, one can calculate the value of T_a given in Corollary 11, and select a valid sampling period T in the range $(0, T_a)$. If the interaction topology among the followers is undirected, all the eigenvalues of D_1 are real and inside the unit circle, then a simplified results can be obtained.

Corollary 13 *Suppose that Assumption 1 holds, and the interaction topology among the followers is undirected. Given positive scalars γ_1 and γ_2, the matrix $\widetilde{A}$ has all eigenvalues within the unit circle if the sampling period $T \in \left(0, \min_{i=1,\dots,m}\left\{\frac{2(1-\lambda_i^2)}{(1-\lambda_i)^2\gamma_2^2 + 2(1-\lambda_i^2)^2\gamma_1}\right\}\right)$.*

2.2.2 Design of scalars γ_1 and γ_2

Though for any given scalars, γ_1 and γ_2, Corollary 12 provides a way to select a valid sampling period T, it is also important to design the scalars, γ_1 and γ_2, under given T. It is because sometimes, small enough sampling period cannot be guaranteed due to economic constraint and energy consumption issues, and each sampling device has limits on its sampling frequency. We provide the following results on the design of γ_1 and γ_2 given any sampling period T.

Proposition 14 *Suppose that Assumption 1 holds. Let λ_i be the ith eigenvalue of D_1. Given the sampling period T, the matrix $\widetilde{A}$ has all eigenvalues within the unit circle if the positive scalars γ_1 and γ_2 are chosen such as*

$$0 < \gamma_1 < \min_{i=1,\ldots,m} \left\{ \frac{|1-\lambda_i|^2}{1-|\lambda_i|^2} \left(\frac{2\gamma_2}{T} - \sqrt{\phi_i} \right) \right\}, \tag{2.13}$$

$$0 < \gamma_2 < \min_{i=1,\ldots,m} \left\{ \frac{2(1-|\lambda_i|^2)|1-\lambda_i|^2}{\left[|1-\lambda_i|^4 + 4\left(\mathrm{Im}\left\{\lambda_i\right\}\right)^2 \right] T} \right\}, \tag{2.14}$$

where $\phi_i = \frac{16\left(\mathrm{Im}\{\lambda_i\}\right)^2 \gamma_2^3}{|1-\lambda_i|^2 T [2(1-|\lambda_i|^2)-|1-\lambda_i|^2 T\gamma_2]}$, $i = 1, \ldots, m$.

Proof: By (2.6), it is easy to see that

$$\gamma_2 < \frac{2(1-|\lambda_i|^2)}{|1-\lambda_i|^2 T}, \quad i = 1, \ldots, m. \tag{2.15}$$

Note that (2.6) is equivalent to

$$\left(\frac{2\gamma_2}{T} - \frac{1-|\lambda_i|^2}{|1-\lambda_i|^2}\gamma_1 \right)^2 - \phi_i < 0, \quad i = 1, \ldots, m, \tag{2.16}$$

where ϕ_i is given in the statement. The left hand side of (2.16) has two zeros, i.e.,

$$\gamma_{11,i}, \text{ and } \gamma_{12,i} = \frac{|1-\lambda_i|^2}{1-|\lambda_i|^2} \left(\frac{2\gamma_2}{T} \pm \sqrt{\phi_i} \right).$$

33

Then the solution to (2.16) is $\gamma_1 > \max_{i=1,\ldots,m} \{\gamma_{11,i}\}$ or $\gamma_1 < \min_{i=1,\ldots,m} \{\gamma_{12,i}\}$. Note that

$\gamma_1 > \max_{i=1,\ldots,m} \{\gamma_{11,i}\}$ contradicts (2.5). Hence, we have (2.13). Since $\gamma_1 > 0$, and in order

to ensure such choice of γ_1 exist, it requires that $\gamma_{12,i} > 0 \; \forall i = 1, \ldots, m$, which yields that

$\gamma_2 < \min_{i=1,\ldots,m} \left\{ \frac{2(1-|\lambda_i|^2)|1-\lambda_i|^2}{\left[|1-\lambda_i|^4+4(\mathrm{Im}\{\lambda_i\})^2\right]T} \right\}$. Combining (2.15), we have (2.14). Therefore, if

γ_1 and γ_2 are selected to respectively satisfy (2.13) and (2.14), (2.5) and (2.6) hold, which

implies that all the eigenvalues of matrix $\widetilde{A}$ are inside the unit circle. $\qquad \square$

It can be seen that the two design parameters γ_1 and γ_2 should be small if the

sampling period T is chosen to be a large number.

In practice, given the graph $\mathcal{G}$ satisfying Assumption 1 and the sampling period T,

one can first choose γ_2 satisfying (2.14), and then choose γ_1 satisfying (2.13) given selected

γ_2. Such choice of γ_1 and γ_2 ensures that (2.5) and (2.6) hold. If the interaction topology

among followers is undirected, a simplified result can be obtained.

Corollary 15 *Suppose that Assumption 1 holds, and the interaction topology among the*

followers is undirected. Given the sampling period T, the matrix $\widetilde{A}$ has all eigenvalues

within the unit circle if the positive scalars γ_1 and γ_2 are respectively chosen such as $0 <$

$\gamma_1 < \min_{i=1,\ldots,m} \left\{ \frac{2(1-\lambda_i)^2\gamma_2}{(1-\lambda_i^2)T} \right\}$ *and* $0 < \gamma_2 < \min_{i=1,\ldots,m} \left\{ \frac{2(1-\lambda_i^2)}{(1-\lambda_i)^2T} \right\}$. $\qquad \square$

Remark 16 *The design of the sampling period T and parameters γ_1 and γ_2 depends on the*

eigenvalues of the matrix D_1, which is related to the underlying interaction graph. In real

applications, one can always let the weights d_{ij}, $j \in \mathcal{N}_i \cup \{i\}$ for agent i to be $\frac{1}{|\mathcal{N}_i|+1}$, which

is valid since it ensures that D is a row-stochastic matrix. The number of possible values

of D such that the underlying interaction graph satisfies Assumption 1, is finite since there

are a finite number of agents. Note that the choice of the sampling period T and the design

parameters γ_1 and γ_2 depends on each other. If the parameters γ_1 and γ_2 are fixed, for each possible D, one can select T by Corollary 12. Among these values of T, the smallest one can be selected for implementation in practice. If the sampling period T is fixed, valid scalars γ_1 and γ_2 can be selected in a similar manner.

2.2.3 Two Special Cases

Discrete-time Single-integrator Agents

When the agents have single-integrator dynamics given by

$$r_i[k+1] = r_i[k] + Tu_i[k], \tag{2.17}$$

we implement the following control law for follower $i \in \mathscr{F}$

$$u_i[k] = \sum_{j \in \mathscr{L} \cup \mathscr{F}} d_{ij} \left(\frac{r_j[k] - r_j[k-1]}{T} - \gamma\{r_i[k] - r_j[k]\} \right), \tag{2.18}$$

where γ is positive constant to be determined. Define the containment control error for follower i as $x_i[k] = \sum_{j=1}^{n} d_{ij}(r_i[k] - r_j[k])$. Define the collective containment control error vector as $X[k] = \left(x_1^\top[k], \dots, x_m^\top[k] \right)^\top$. Let $W[k] = \left(X^\top[k+1], X^\top[k] \right)^\top$. Then we have

$$W[k+1] = \widetilde{A}_1 W[k] + \widetilde{B}_1 \Delta_L[k], \tag{2.19}$$

where $\widetilde{A}_1 = A_1 \otimes I_p$ and $\widetilde{B}_1 = \begin{bmatrix} D_2 \otimes I_p \\ \\ \mathbf{0}_{mp \times (n-m)p} \end{bmatrix}$ with

$$A_1 = \begin{bmatrix} (1 - T\gamma)I_m + (1 + T\gamma)D_1 & -D_1 \\ \\ I_m & \mathbf{0}_{m \times m} \end{bmatrix}. \tag{2.20}$$

35

Lemma 17 *Suppose that Assumption 1 holds. Let λ_i be the ith eigenvalue of D_1. Then $\theta_i > 0$ holds, where $\theta_i = \frac{2|1-\lambda_i|^2[2(1-\mathrm{Re}\{\lambda\}_i)-|1-\lambda_i|^2]}{|1-\lambda_i|^4+4[\mathrm{Im}\{\lambda_i\}]^2}$. If the positive scalars T and γ satisfy*

$$T\gamma < \min\left\{1, \min_{i=1,\dots,m} \theta_i\right\}, \tag{2.21}$$

then the matrix $\widetilde{A}_1$ has all eigenvalues within the unit circle.

Proof: The proof can be derived by following a similar analysis of Lemma 3.3 in [10] and the properties of the Kronecker product, thus is omitted here. $\square$

Corollary 18 *Suppose that Assumptions 1 holds and the leaders' inputs are bounded, i.e., $\|u_i[k]\|_2 \le c_3$, $i \in \mathscr{L}$, where c_3 is a positive constant. If the positive scalars T and γ satisfy (2.21), using the algorithms (2.18) for (2.17), the followers converge to the convex hull spanned by the leaders with bounded position containment error, and the ultimate bound of overall containment control error, $\|X[k]\|_2$, is $c_3 c_4 T \sqrt{2(n-m)} \left\|\widetilde{B}_1\right\|_2 /(1-\rho_1)$, where positive constant c_4 and $\rho_1 \in [0,1)$ satisfy $\left\|\widetilde{A}_1^j\right\|_2 \le c_4 \rho_1^j$, $j \ge 0$.*

Proof: By following a similar analysis in the proof of Theorem 7, it can be obtained that $\lim_{k\to\infty} \|W[k]\|_2 \le 2c_3 c_4 T \sqrt{n-m} \left\|\widetilde{B}_1\right\|_2 /(1-\rho_1)$. Therefore, it holds that $\lim_{k\to\infty} \|X[k]\|_2 = \frac{1}{2}\lim_{k\to\infty} \sqrt{(\|X[k]\|_2 + \|X[k-1]\|_2)^2} \le \frac{\sqrt{2}}{2}\lim_{k\to\infty} \|W[k]\|_2 \le c_3 c_4 T \sqrt{2(n-m)} \left\|\widetilde{B}_1\right\|_2 /(1-\rho_1)$. $\square$

Discrete-time Double-integrator Agents

When the agent i's model is descretized as

$$r_i[k+1] = r_i[k] + Tv_i[k]$$

$$v_i[k+1] = v_i[k] + Tu_i[k] \tag{2.22}$$

where $r_i[k] \in \mathbb{R}^p$, $v_i[k] \in \mathbb{R}^p$ and $u_i[k] \in \mathbb{R}^p$ represent the position, velocity, and control input of the ith agent at $t = kT$, respectively.

Use the same definitions of $x_i[k]$ and $y_i[k]$ respectively for the position and velocity containment errors of the follower i, and $X[k]$ and $Y[k]$ respectively for the collective position and velocity containment control errors. Using (3.5) for (2.22) for each follower, and defining $Z[k+1] = \left(X^\top[k+1], Y^\top[k+1], X^\top[k], Y^\top[k]\right)^\top$, we then have the same form of system as

$$Z[k+1] = \widetilde{A}_2 Z[k] + \widetilde{B}_2 \Delta[k],$$

with $\widetilde{A}_2 = A_2 \otimes I_p$ and $\widetilde{B}_2 = [1, 1, 0, 0]^\top \otimes \left(D_2 \otimes I_p\right)$, where

$$A_2 = \begin{bmatrix} I_m & TI_m & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} \\ \bar{A}_{21} & \bar{A}_{22} & \mathbf{0}_{m\times m} & -D_1 \\ I_m & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} \\ \mathbf{0}_{m\times m} & I_m & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} \end{bmatrix}$$

with $\bar{A}_{21} = -T\gamma_1(I_m - D_1)$ and $\bar{A}_{22} = (1 - T\gamma_2)I_m + (1 + T\gamma_2)D_1$.

Lemma 19 *Suppose that Assumption 1 holds. Let λ_i be the ith eigenvalue of D_1. The matrix $\widetilde{A}_2$ has all eigenvalues within the unit circle if and only if there exist positive scalars T, γ_1 and γ_2 such that*

$$\frac{2\gamma_2}{\gamma_1} > \left(\frac{1 - |\lambda_i|^2}{|1 - \lambda_i|^2} + 1\right)T, \quad i = 1, \dots, m, \tag{2.23}$$

and

$$\left[\frac{1}{T\gamma_1} - \frac{1 - |\lambda_i|^2}{|1 - \lambda_i|^2(2\gamma_2 - T\gamma_1)}\right]^2 \left[\frac{4(1 - |\lambda_i|^2)}{|1 - \lambda_i|^2(2T\gamma_2 - T^2\gamma_1)}\right.$$
$$\left. - 1\right] - \frac{16(\operatorname{Im}\{\lambda_i\})^2}{|1 - \lambda_i|^4 T^2\gamma_1^2} > 0, \quad i = 1, \dots, m. \tag{2.24}$$

37

In addition, such scalars T, γ_1 and γ_2 always exist.

Proof: The proof can be obtained by following a similar analysis procedure in the proof of Lemma 6, thus is omitted here. □

Corollary 20 *Suppose that Assumptions 1 and 4 hold. If the positive scalars T, γ_1 and γ_2 satisfy (2.23) and (2.24), the followers converge to the convex hull spanned by the leaders with bounded position and velocity containment error, and the ultimate bound of overall containment control error, $\|X[k]\|_2 + \|Y[k]\|_2$, is $2c_1c_5T\sqrt{n-m}\left\|\widetilde{B}_2\right\|_2/(1-\rho_2)$, where c_1 is given in Assumption 4, and positive constants c_5 and $\rho_2 \in [0,1)$ satisfy $\left\|\widetilde{A}_2^j\right\|_2 \le c_5\rho_2^j$, $j \ge 0$.* □

Remark 21 *The containment control problem for agents with the same model as (2.22) has been addressed in [2]. However, in [2], the leaders' dynamics are assumed to be the same as the followers with zero control inputs. The proposed algorithm (3.5) can deal with the case where the leaders have bounded nonzero inputs, which is more general, and the corresponding result takes into account the more realistic sampled-data setting.*

2.3 An Illustrative Example

We provide a simulation to illustrate the results obtained in previous section.

Consider a group of ten agents, which are labeled as $1, \ldots, 10$. Denote by $\mathscr{F} = \{1, \ldots, 6\}$ and $\mathscr{L} = \{7, \ldots, 10\}$ the sets of followers and leaders, respectively. The directed communication network is shown in Fig. 4.1. Let $(r_{x_i}[k], r_{y_i}[k])$ and $(v_{x_i}[k], v_{y_i}[k])$ be the coordinates of agent i's position and velocity at time k. The input of the ith leaders is

chosen to be $u_i[k] = -\frac{1}{(i-6)^2}\sin(\frac{1}{i-6}k) + 0.01(i-6)^2 e^{-0.1(i-6)k}$, $i \in \mathcal{L}$. Set the sampling

period T to 0.1, and choose $\gamma_1 = 0.9$ and $\gamma_2 = 1.25$. The control law (3.5) is implemented for

all the followers with dynamics (2.1). The resulting trajectories of positions and velocities

are shown in Fig. 2.2. It can be seen that both positions and velocities of all the followers

converge to the convex hull spanned by those of the four leaders.
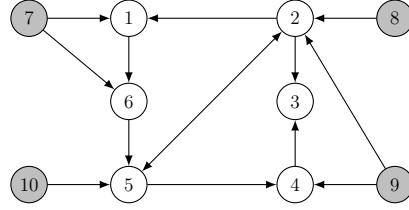
Figure 2.1: Directed network topology for a group of ten agents, which are labeled from 1 to 10. There are four leaders, which are denoted by grey-filled circles. The rest are the followers.
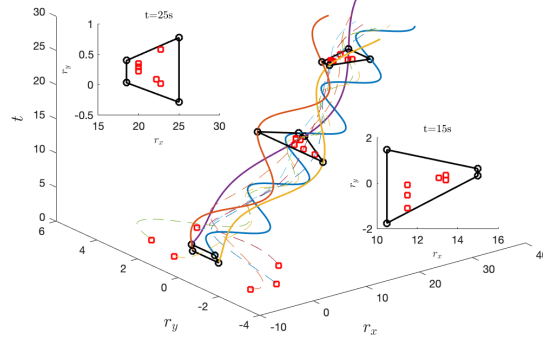
(a) Position.
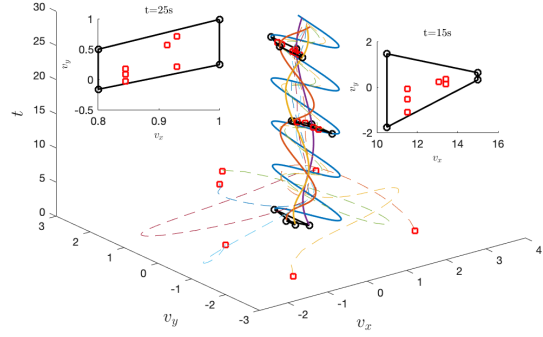
(b) Velocity.

Figure 2.2: Position and velocity trajectories of the agents with dynamics (2.1) under a directed network topology presented in Fig. 4.1. The followers' input is implemented with (3.5). The solid lines denote the trajectories of leaders' positions and velocities, and the dashed lines denote the followers' positions and velocities. The black circles and red squares denote the positions (velocities) of the leaders and followers, respectively. The areas formed by four connecting black lines are convex hull spanned by the leaders. Two snapshots at $t = 15s$ and $t = 25s$ show that all the followers' positions and velocities are in the convex hull spanned by the leaders.

Chapter 3

Robust Distributed Average Tracking for Double-integrator Agents Without Velocity Measurements Under Event-triggered Communication

3.1 Problem Statement

In this chapter, we consider N physical agents, and the interaction topology among these agents is characterized as the undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Unless otherwise stated, throughout this chapter, we assume a time-invariant graph. Each agent i is modeled by

double-integrator dynamics

$$\dot{x}_i(t) = v_i(t), \quad \dot{v}_i(t) = u_i(t), \quad i \in \mathcal{V}, \tag{3.1}$$

where $x_i(t) \in \mathbb{R}^p$ and $v_i(t) \in \mathbb{R}^p$ are the ith agent's position and velocity, respectively, and u_i is its control input.

Each agent has a time-varying reference signal $x_i^r \in \mathbb{R}^p$, $i \in \mathcal{V}$ satisfying

$$\dot{x}_i^r(t) = v_i^r(t), \quad \dot{v}_i^r(t) = u_i^r(t), \quad i \in \mathcal{V}, \tag{3.2}$$

where $v_i^r(t) \in \mathbb{R}^p$ and $u_i^r(t) \in \mathbb{R}^p$ are the velocity and acceleration of the ith agent's reference signal, respectively. We assume that the reference signals are generated internally by the agents, and that each agent has access to its own reference signal, and the velocity and acceleration of the reference signal. In this chapter, we make the following assumption on the reference signals, and the velocities and accelerations of the reference signals.

Assumption 3 *For any two connected agents, the local difference in reference signals $x_i^r(t)$, their velocities $v_i^r(t)$ and their accelerations $a_i^r(t)$ are bounded, i.e., $\sup_{\substack{t \in [0,\infty) \\ \forall (i,j) \in \mathcal{E}}} \left\| x_i^r(t) - x_j^r(t) \right\|_\infty \leq \bar{x}^r$, $\sup_{\substack{t \in [0,\infty) \\ \forall (i,j) \in \mathcal{E}}} \left\| v_i^r(t) - v_j^r(t) \right\|_\infty \leq \bar{v}^r$, and $\sup_{\substack{t \in [0,\infty) \\ \forall (i,j) \in \mathcal{E}}} \left\| u_i^r(t) - u_j^r(t) \right\|_\infty \leq \bar{a}^r$.*

In the distributed average tracking for a group of double-integrator agents, the objective is to design controller u_i for agent $i \in \mathcal{V}$ such that each agent's position (velocity) is capable of tracking the group average of their reference signals (their reference signals' velocity). That is, for any $i \in \mathcal{V}$, it is achieved that $\lim_{t\to\infty} \left\| x_i(t) - \frac{1}{N}\sum_{j=1}^N x_j^r(t) \right\|_2 = 0$ and $\lim_{t\to\infty} \left\| v_i(t) - \frac{1}{N}\sum_{j=1}^N v_j^r(t) \right\|_2 = 0$. In this chapter, we are particularly interested in developing a controller for each agent without velocity measurement and in the absence of any correct initialization. The motivation behind is that employing velocity measuring

42

device is usually costly in the aspect of finance and energy. Also, the velocity measurements are less accurate compared with position measurements. On the other hand, perfect initialization is hard to achieve in reality.

Before moving onto the main results, a lemma is presented in the following.

Lemma 22 *[7] For any symmetric real matrix, M, of the form $M = \begin{bmatrix} D_{11} & D_{12} \\ D_{12}^T & D_{22} \end{bmatrix}$, it holds that $M \succ 0$ if and only if one of the following condition hold: i) $D_{11} \succ 0$ and $D_{22} - D_{12}^T D_{11}^{-1} D_{12} \succ 0$; ii) $D_{22} \succ 0$ and $D_{11} - D_{12} D_{11}^{-1} D_{12}^T \succ 0$.*

3.2 Distributed Average Tracking without Velocity Measurements

In this section, we introduce a distributed average tracking algorithm for double-integrator agents without using the velocity measurements and in the absence of any correct initialization. In the rest of the chapter, we omit the argument t for brevity.

We design a filter for each agent i as

$$\dot{\phi}_i = -\kappa(x_i - x_i^r) - 2\kappa(w_i - v_i^r) + u_i^r$$

$$- \sum_{j=1}^{N} a_{ij} \pi_{ij} \mathrm{sgn}(x_i - x_j + w_i - w_j)$$

$$w_i = \phi_i + \kappa(x_i - x_i^r), \quad i \in \mathcal{V}, \tag{3.3}$$

where $\kappa \in \mathbb{R}$ is a positive constant to be determined, $\phi_i \in \mathbb{R}^p$ is the internal state of the filter, $w_i \in \mathbb{R}^p$ is the output of the filter, and π_{ij} is a time-varying gain for the edge $(i, j) \in \mathcal{E}$,

satisfying the following adaptation law

$$\dot{\pi}_{ij} = a_{ij} \left\| x_i - x_j + w_i - w_j \right\|_1, \quad i \in \mathcal{V} \tag{3.4}$$

with $\pi_{ij}(0) > 0$ if $(i,j) \in \mathcal{E}$. In addition, each agent i needs to coordinate with its neighbor $j \in \mathcal{N}_i$ to ensure $\pi_{ij}(0) = \pi_{ji}(0)$. In this way, the gains π_{ij} and π_{ji} remain equal to each other. We design the controller for agent i as

$$
\begin{aligned}
u_i = & -\kappa(x_i - x_i^r) - \kappa(w_i - v_i^r) + u_i^r \\
& - \sum_{j=1}^{N} a_{ij} \pi_{ij} \mathrm{sgn}(x_i - x_j + w_i - w_j), \quad i \in \mathcal{V}.
\end{aligned}
\tag{3.5}
$$

Essentially, the filter is designed such that its output is capable of tracking the average of the reference signals' velocities, and the controller is applied to drive each agent's position to the average of the reference signals and velocity to the output of the filter. Note that the designs of the filter (3.3) and the controller (3.5) for each agent i depend on only local information and the positions and filter's outputs from its neighbors. Therefore, it is implementable in reality.

Remark 23 *Note that there is no requirement on the initialization of each agents' position and velocity, as well as the internal state of the filter. Thus, the proposed algorithm (3.3)-(3.5) is called robust distributed average tracking algorithm.*

Let $x = \left[x_1^T, \ldots, x_N^T\right]^T$, $v = \left[v_1^T, \ldots, v_N^T\right]^T$, and $w = \left[w_1^T, \ldots, w_N^T\right]^T$. Define $\tilde{x} = (M \otimes I_p)x$, $\tilde{v} = (M \otimes I_p)v$, and $\tilde{w} = (M \otimes I_p)w$, where $M = I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T$. For brevity,

define $\alpha = \left[\alpha_1^T, \ldots, \alpha_N^T\right]^T$ with $\alpha_i = \kappa x_i^r + \kappa v_i^r + u_i^r$. Then we have

$$\dot{\tilde{x}} = \tilde{v}$$

$$\dot{\tilde{v}} = -\kappa\tilde{x} - \kappa\tilde{w} + (M \otimes I_p)\alpha$$

$$- (B\Pi \otimes I_p)\text{sgn}\left[\left(B^T \otimes I_p\right)(\tilde{x} + \tilde{w})\right], \tag{3.6}$$

and

$$\dot{\tilde{w}} = -\kappa\tilde{x} - 2\kappa\tilde{w} + \kappa\tilde{v} + (M \otimes I_p)\alpha$$

$$- (B\Pi \otimes I_p)\text{sgn}\left[\left(B^T \otimes I_p\right)(\tilde{x} + \tilde{w})\right], \tag{3.7}$$

where $\Pi \in \mathbb{R}^{|\mathcal{E}| \times |\mathcal{E}|}$ is a time-varying diagonal matrix, and the sth diagonal entry, denoted by Π_{ss}, represents the weight on the sth edge. That is, if the sth edge is between agent i and agent j, then $\Pi_{ss} = \pi_{ij}$.

Theorem 24 *Suppose that the undirected graph $\mathcal{G}$ is connected, and Assumption 4 holds. Using the algorithm (3.3)-(3.5) for (3.1), distributed average tracking is achieved asymptotically if $\kappa > \frac{3+2\sqrt{3}}{3}$.*

Proof: We prove this statement in two steps. In the first step, we prove that for any $i \in \mathcal{V}$, $x_i \to \frac{1}{N}\sum_{j=1}^N x_j$ and $v_i \to \frac{1}{N}\sum_{j=1}^N v_j$ as $t \to \infty$. In the second step, we prove that for any $i \in \mathcal{V}$, $\sum_{j=1}^N x_j \to \sum_{j=1}^N x_j^r$ and $\sum_{j=1}^N v_j \to \sum_{j=1}^N v_j^r$ as $t \to \infty$. Combining these two steps, it can be concluded that $\lim_{t\to\infty}\left\|x_i - \frac{1}{N}\sum_{j=1}^N x_j^r\right\|_2 = 0$ and $\lim_{t\to\infty}\left\|v_i - \frac{1}{N}\sum_{j=1}^N v_j^r\right\|_2 = 0$ hold for all $i \in \mathcal{V}$. For simplicity, we denote these two steps by consensus and sum-tracking steps, respectively.

Define $X = \left[\tilde{x}^T, \tilde{v}^T, \tilde{w}^T\right]^T$. Consider a Lyapunov function candidate as

$$V = \frac{1}{2}X^T P X + \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{(\pi_{ij} - \pi_m)^2}{4}, \tag{3.8}$$

where

$$P = \begin{bmatrix} \mu I_{Np} & \mathbf{0}_{Np \times Np} & I_{Np} \\ \mathbf{0}_{Np \times Np} & I_{Np} & -I_{Np} \\ I_{Np} & -I_{Np} & 2I_{Np} \end{bmatrix}, \tag{3.9}$$

and π_m is a positive constant to be determined. By Lemma 22 and the properties of the Kronecker product, it holds that P is positive definite if and only if $\mu > 1$. Therefore, V is positive definite.

Taking the derivative of V along (3.6)-(3.7) yields

$$\dot{V} = -X^T Q X + (\tilde{x} + \tilde{w})^T (M \otimes I_p)\alpha$$
$$- \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\pi_{ij} \left\| x_i - x_j + w_i - w_j \right\|_1$$
$$+ \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \pi_{ij}\dot{\pi}_{ij} - \frac{\pi_m}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \dot{\pi}_{ij}$$

where

$$Q = \begin{bmatrix} \kappa I_{Np} & -\frac{\mu+\kappa}{2}I_{Np} & \frac{3\kappa}{2}I_{Np} \\ -\frac{\mu+\kappa}{2}I_{Np} & \kappa I_{Np} & -\frac{1+3\kappa}{2}I_{Np} \\ \frac{3\kappa}{2}I_{Np} & -\frac{1+3\kappa}{2}I_{Np} & 3\kappa I_{Np} \end{bmatrix}. \tag{3.10}$$

Note that $\|\alpha_i - \alpha_j\|_\infty \leq \bar{\alpha}$ by Assumption 4, and let $N_{\max} = \max_{i \in \mathcal{V}} |\mathcal{N}_i|$. Then it holds that

$$\|(M \otimes I_p)\alpha\|_\infty \leq \frac{1}{N}\max_{i \in \mathcal{V}}\left\{\sum_{j=1, j\neq i}^{N}\|\alpha_i - \alpha_j\|_\infty\right\}$$
$$\leq \frac{N-1}{2N}\sum_{i=1}^{N}\sum_{j \in \mathcal{N}_i}\|\alpha_i - \alpha_j\|_\infty \leq \frac{\bar{\alpha}N_{\max}(N-1)}{2}, \tag{3.11}$$

46

where $\bar{\alpha} = \kappa \bar{x}^r + \kappa \bar{v}^r + \bar{a}^r$. For brevity, define

$$\beta = \frac{\bar{\alpha} N_{\max}(N-1)}{2}. \tag{3.12}$$

Note that

$$\|\tilde{x} + \tilde{w}\|_1 \leq \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1, j \neq i} \|x_i - x_j + w_i - w_j\|_1$$

$$\leq \max_{i \in \mathcal{V}} \left\{ \sum_{j=1, j \neq i}^{N} \|x_i - x_j + w_i - w_j\|_1 \right\}$$

$$\leq \frac{N-1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \|x_i - x_j + w_i - w_j\|_1.$$

It then holds that $(\tilde{x}+\tilde{w})^T (M \otimes I_p)\alpha \leq \frac{(N-1)\beta}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \|x_i - x_j + w_i - w_j\|_1$. Then, it

follows that $\dot{V} \leq -X^T Q X - \frac{\pi_m - (N-1)\beta}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \|x_i - x_j + w_i - w_j\|_1$, where the fact

that $(\tilde{x}+\tilde{w})^T (B\Pi \otimes I_p)\text{sgn}[(B^T \otimes I_p)(\tilde{x}+\tilde{w})] = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij}\pi_{ij} \|(x_i - x_j) + (w_i - w_j)\|_1$

is used. Selecting an π_m such that $\pi_m \geq \beta$, one has

$$\dot{V} \leq -X^T Q X := -W[X].$$

By Lemma 22, the matrix Q is positive definite if and only if $\kappa > \mu + \frac{1}{3(\mu-1)} = f(\mu)$,

which implies that Q is positive definite if $\kappa > \min_{\mu>1} f(\mu) = \frac{3+2\sqrt{3}}{3}$. Thus $\dot{V} \leq 0$,

which implies that V is nonincreasing. Then it follows that X and π_{ij} are bounded. Note

that V is bounded from below by zero. Thus, $\lim_{t \to \infty} V$ exists and is finite. Note that

$\int_0^t W[X(\tau)]d\tau \leq -\int_0^t \dot{V}[X(\tau), \{\pi_{ij}(\tau)\}_{i,j \in \mathcal{V}}]d\tau = V[X(0), \{\pi_{ij}(0)\}_{i,j \in \mathcal{V}}] - V[X, \{\pi_{ij}\}_{i,j \in \mathcal{V}}]$.

Therefore, $\lim_{t \to \infty} \int_0^t W[X(\tau)]d\tau$ exists and is finite. It follows from (3.6), (3.7) and As-

sumption 4 that $\dot{\tilde{x}}$, $\dot{\tilde{v}}$ and $\dot{\tilde{w}}$ are bounded. Hence, $\tilde{x}$, $\tilde{v}$, and $\tilde{w}$ are uniformly continuous. Con-

sequently, $W[X]$ is uniformly continuous by the definition of $W[X]$ and X. By Barbalat's

Lemma, it can be concluded that $W[X] \to 0$ as $t \to \infty$, which implies that $\lim_{t \to \infty} X = \mathbf{0}_{np}$. This completes the consensus step.

Second, define $S_x = \sum_{j=1}^{N} x_j - \sum_{j=1}^{N} x_j^r$, $S_v = \sum_{j=1}^{N} v_j - \sum_{j=1}^{N} v_j^r$, and $S_w = \sum_{j=1}^{N} w_j - \sum_{j=1}^{N} v_j^r$. Then we have that $\dot{S} = \left(\begin{bmatrix} 0 & 1 & 0 \\ -\kappa & 0 & -\kappa \\ -\kappa & \kappa & -2\kappa \end{bmatrix} \otimes I_p \right) S = (A \otimes I_p)S,$

where $S = [S_x^T, S_v^T, S_w^T]^T$. The characteristic polynomial of A is $p_A(s) = s^3 + 2\kappa s^2 + (\kappa + \kappa^2)s + \kappa^2$. According to the Routh-Hurwitz stability criterion, it is easy to verify that if $\kappa > 0$, all the zeros of $p_A(s) = 0$ have negative real parts, which means that A is Hurwitz. Note that $\kappa > \frac{3+2\sqrt{3}}{3} > 0$. Then the matrix A is Hurwitz, which indicates $\lim_{t \to \infty} S = \mathbf{0}_{3p}$. This completes the sum-tracking step. $\blacksquare$

Note that the dynamics (3.6) is discontinuous due to the introduction of the signum function in the controller and filter design (3.3)-(3.5). Then, the solutions should be understood in terms of differential inclusion by using non-smooth analysis [37, 26]. However, since the signum function is measurable and locally essentially bounded, the Filippov solutions for the closed-loop dynamics always exist. The Lyapunov function used in the proof is continuously differentiable. Then its set-valued Lie derivative is a singleton at the discontinuous points. Therefore, the proof is valid as in the case without discontinuities.

Remark 25 *Note that the algorithm (3.3)-(3.5) has some connection with the first algorithm in [50]. In the first algorithm in [50], there are multiple design parameters, the design of which depends on the largest and smallest nonzero eigenvalues of the Laplacian matrix, the bounds on the reference signals, and the total number of the agents in the network. Also, the first algorithm in [50] is sensitive to parameter selection as a certain parameter is*

required to be exactly equal to a certain value. However, the algorithm (3.3)-(3.5) overcomes these limitations in [50], and solve the distributed average tracking problem if κ is greater than a constant. It is easy to select a suitable value for κ and implement the algorithm. It is also worth noting that the structures of the controller (3.5) and the filter (3.3) are different from the ones in [50]. Such newly designed structures and their independence of global information lay a solid base for the development of event-triggered approaches.

Remark 26 *The algorithm (3.3)-(3.5) is implementable since κ is constant, which can be chosen off-line before running the algorithm and embedded to each agent. Once the algorithm starts to run, the agents communicate with only local neighbors and there is no need to have access to any global information. If each agent chooses its own $\kappa_i(0)$ off-line such that $\kappa_i(0) > \frac{3+2\sqrt{3}}{3}$, then each agent can run the max consensus algorithm in [51]: $\kappa_i(k+1) = \max_{j \in \mathcal{N}_i \cup \{i\}} \{\kappa_j(k)\}$, where k is discrete time instance, to drive each agent to reach consensus on $\max_{j \in \mathcal{V}} \kappa_j(0)$. It is proved that the max consensus algorithm converges in finite time. To determine when to stop the max consensus algorithm, each agent needs to know the diameter of the graph. However, one can always be more conservative to run the max consensus algorithm long enough, which guarantees the convergence.*

Remark 27 *Theorem 38 shows that the agents are capable of achieving distributed average tracking under any fixed connected undirected communication network. It is actually able to extend to the case of arbitrarily switching connected communication networks with positive dwelling time. The function defined in (3.8) can be used as a common Lyapunov function during the proof process.*

3.3   Event-triggered Distributed Average Tracking without Velocity Measurements

The algorithm (3.3)-(3.5) in Section 3.2 requires each agent i to continuously exchange the position, x_i, and the output of the filter, w_i, with its neighbors. However, continuous communication may not be practical due to the constrained bandwidth of the communication network in reality. To this end, we investigate the event-triggered distributed average tracking, which removes the requirement of continuous communication. It is worth mentioning that no velocity measurements and no initialization requirements are needed as well.

It is noted that there are several practical limitations for the event-triggered algorithm in [45]. First, due to the nature of the adaptation law, the adaptive gains can only increase. It is normally the case that there exist measurement/communication noise and/or persistent disturbances in practical systems. In such case, perfect consensus cannot be achieved, and consequently, the adaptive gains and the control inputs will grow unbounded, which would affect the convergence and the success of the event-triggered scheme. Second, implementing the algorithm in [45] requires each agent to maintain an additional internal dynamics to ensure the exclusion of Zeno behavior. Such additional dynamics may cost extra computational power and storage space. Finally, the use of the signum function in the algorithm design will cause chattering phenomenon in real applications. To overcome these limitations, we propose a novel event-triggered distributed average tracking algorithm without using velocity measurements and requiring correct initialization.

We propose the following distributed average tracking algorithm with the filter

$$\dot{\phi}_i = -\kappa(x_i - x_i^r) - 2\kappa(w_i - v_i^r) + u_i^r$$

$$- \sum_{j=1}^{N} a_{ij}\pi_{ij}h(\hat{x}_i - \hat{x}_j + \hat{w}_i - \hat{w}_j, t)$$

$$w_i = \phi_i + \kappa(x_i - x_i^r), \quad i \in \mathcal{V}, \tag{3.13}$$

and the controller

$$u_i = -\kappa(x_i - x_i^r) - \kappa(w_i - v_i^r) + u_i^r$$

$$- \sum_{j=1}^{N} a_{ij}\pi_{ij}h(\hat{x}_i - \hat{x}_j + \hat{w}_i - \hat{w}_j, t), \quad i \in \mathcal{V}, \tag{3.14}$$

and π_{ij} is governed by the following adaptation law

$$\dot{\pi}_{ij} = a_{ij}\big[-\rho_{ij}\pi_{ij} + R_i + \big(\hat{x}_i - \hat{x}_j + \hat{w}_i - \hat{w}_j\big)^T$$

$$\times h\big(\hat{x}_i - \hat{x}_j + \hat{w}_i - \hat{w}_j, t\big)\big], \quad i \in \mathcal{V}, \tag{3.15}$$

where $\hat{x}_j(t) = x_j(t_{k_j}^j)$ and $\hat{w}_j(t) = w_j(t_{k_j}^j)$ $t \in [t_{k_j}^j, t_{k_j+1}^j)$, denote the last broadcast position and filter output of agent j, respectively, and $t_{k_j}^j = \max\{t_k^j \mid t_k^j \leq t\}$ is the latest triggering time instant of agent j, ρ_{ij} and R_i are positive constants to be determined, and $h : \mathbb{R}^p \times \mathbb{R}_{\geq 0} \to \mathbb{R}^p$ is a nonlinear function [34] defined as

$$h(z, t) = \frac{z}{\|z\|_2 + \eta e^{-ct}},$$

where η and c are positive constants. The boundary layer ηe^{-ct} is time varying, and as $t \to \infty$, the continuous function $h(z, t)$ approaches the discontinuous function $\text{sgn}(z)$.

For each agent $i \in \mathcal{V}$, define

$$e_{x_i} = \hat{x}_i - x_i, \quad e_{w_i} = \hat{w}_i - w_i, \tag{3.16}$$

51

and the triggering time instant is determined by $t_1^i = 0$ and

$$t_{k+1}^i = \min \left\{ t \mid f_i \left(t, x_i, w_i, \{\hat{x}_j, \hat{w}_j\}_{j \in \mathcal{N}_i \cup \{i\}} \right) > 0 \right\}, \tag{3.17}$$

where $f_i \left(t, x_i, w_i, \{\hat{x}_j, \hat{w}_j\}_{j \in \mathcal{N}_i \cup \{i\}} \right)$ is agent i's triggering function, which is given by

$$
\begin{aligned}
& f_i \left(t, x_i, w_i, \{\hat{x}_j, \hat{w}_j\}_{j \in \mathcal{N}_i \cup \{i\}} \right) \\
&= \left| \|e_{x_i} + e_{w_i}\|_1 R_i + (e_{x_i} + e_{w_i})^T \hat{\zeta}_i \right| - \epsilon_i e^{-\varphi_i t},
\end{aligned}
\tag{3.18}
$$

where $\hat{\zeta}_i = \sum_{j=1}^N a_{ij} \pi_{ij} h(\hat{x}_i - \hat{x}_j + \hat{w}_i - \hat{w}_j, t)$, and R_i, ϵ_i and φ_i are positive constants to be determined. Note that the triggering function in (3.18) takes values in $\mathbb{R}$ and depends on time t, its current position x_i and current filter's output w_i, and its own and neighbors' last broadcast positions $\{\hat{x}_j\}_{j \in \mathcal{N}_i \cup \{i\}}$ and filter's outputs $\{\hat{w}_j\}_{j \in \mathcal{N}_i \cup \{i\}}$. For agent i, at the triggering time instant, it updates its filter's input and controller by using its current position and filter's output, and broadcasts its current position and filter's output to its neighbors. In the meantime, e_{x_i} and e_{w_i} are reset to zero. When an event is triggered at its neighboring agent j, it receives newly broadcast position and filter's output, and update its filter's input and controller immediately.

Theorem 28 *Suppose that the undirected graph $\mathcal{G}$ is connected, and Assumption 4 holds. Apply the algorithm (3.13)-(3.15) to (3.1) with $\kappa > \frac{3 + 2\sqrt{3}}{3}$, and the triggering time instant is determined by (3.17) with the triggering function defined in (3.18), where $\rho_{ij} > \frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)}$, $\epsilon_i > 0$, $\varphi_i > 0$, $\eta > 0$, $c > 0$, and the matrices P and Q are given in (3.9) and (3.10) with $\mu = \frac{3 + \sqrt{3}}{3}$, respectively. Then,*

(i) if $\beta \le R_i < \beta \max_{j \in \mathcal{N}_i} \{1, \rho_{ij} \sqrt{p}(N-1)\}$, distributed average tracking is achieved with bounded error;

(ii) if $R_i \geq \beta \max_{j \in \mathcal{N}_i}\{1, \rho_{ij}\sqrt{p}(N-1)\}$, distributed average tracking is achieved with zero

　　error.

In addition, the triggering law (3.17) excludes Zeno behavior while running the algorithm (3.13)-(3.15)

　　Proof: We first prove statement (i). The proof follows the same two steps described in that of Theorem 38. Use the same definitions of $\tilde{x}$, $\tilde{v}$ and $\tilde{w}$ as in Section 3.2. For notational simplicity, let $\chi = \tilde{x} + \tilde{w}$ and $\hat{\chi} = (M \otimes I_p)(\hat{x} + \hat{w})$ with $\chi_i = \tilde{x}_i + \tilde{w}_i = x_i - \frac{1}{N}\sum_{j=1}^{N} x_j + w_i - \frac{1}{N}\sum_{j=1}^{N} w_j$ and $\hat{\chi}_i = \hat{x}_i - \frac{1}{N}\sum_{j=1}^{N}\hat{x}_j + \hat{w}_i - \frac{1}{N}\sum_{j=1}^{N}\hat{w}_j$. Then we have

$$\dot{\tilde{x}} = \tilde{v}$$

$$\dot{\tilde{v}} = -\kappa\tilde{x} - \kappa\tilde{w} + (M \otimes I_p)\alpha$$

$$- \begin{bmatrix} \sum_{j=1}^{N} a_{1j}\pi_{1j}h(\hat{\chi}_1 - \hat{\chi}_j, t) \\ \vdots \\ \sum_{j=1}^{N} a_{Nj}\pi_{Nj}h(\hat{\chi}_N - \hat{\chi}_j, t) \end{bmatrix} \tag{3.19}$$

Consider the function V defined in (3.8). Taking the derivative of V along (3.19) yields

$$\dot{V} = -X^T Q X + \chi^T(M \otimes I_p)\alpha$$

$$- \chi^T \begin{bmatrix} \sum_{j=1}^{N} a_{1j}\pi_{1j}h(\hat{\chi}_1 - \hat{\chi}_j, t) \\ \vdots \\ \sum_{j=1}^{N} a_{Nj}\pi_{Nj}h(\hat{\chi}_N - \hat{\chi}_j, t) \end{bmatrix}$$

$$+ \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \dot{\pi}_{ij}(\pi_{ij} - \pi_m).$$

Then by using the facts that $\pi_{ij} = \pi_{ji}$ and $h(-z,t) = -h(z,t)$, it holds that

$$\dot{V} \le -X^T Q X - (e_x + e_w)^T (M \otimes I_p)\alpha$$

$$+ \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\bar{\beta}}{2} \|\hat{\chi}_i - \hat{\chi}_j\|_1$$

$$+ \sum_{i=1}^{N} (e_{x_i} + e_{w_i})^T \sum_{j=1}^{N} a_{ij}\pi_{ij} h(\hat{x}_i - \hat{x}_j + \hat{w}_i - \hat{w}_j, t)$$

$$+ \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{2} \left(-\pi_{ij} + \frac{R_i}{\rho_{ij}}\right)(\pi_{ij} - \pi_m)$$

$$- \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\pi_m}{2}(\hat{\chi}_i - \hat{\chi}_j)^T h(\hat{\chi}_i - \hat{\chi}_j, t),$$

where $\bar{\beta} = (N-1)\beta$. Note that

$$\sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \left[\frac{\bar{\beta}}{2} \|\hat{\chi}_i - \hat{\chi}_j\|_1 - \frac{\pi_m}{2}(\hat{\chi}_i - \hat{\chi}_j)^T h(\hat{\chi}_i - \hat{\chi}_j, t)\right]$$

$$= \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \left(\frac{\bar{\beta}}{2} \|\hat{\chi}_i - \hat{\chi}_j\|_1 - \frac{\pi_m}{2} \frac{a_{ij} \|\hat{\chi}_i - \hat{\chi}_j\|_2^2}{\|\hat{\chi}_i - \hat{\chi}_j\|_2 + \eta e^{-ct}}\right)$$

$$\le \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \left(\frac{\bar{\beta}\sqrt{p} - \pi_m}{2} \|\hat{\chi}_i - \hat{\chi}_j\|_2 + \frac{\pi_m}{2}\eta e^{-ct}\right),$$

where the fact that $\|\hat{\chi}_i - \hat{\chi}_j\|_1 \le \sqrt{p} \|\hat{\chi}_i - \hat{\chi}_j\|_2$ is used. Since $ab \le \frac{\epsilon a^2}{2} + \frac{b^2}{2\epsilon}$ $\forall a, b \in \mathbb{R}$ holds

for any $\epsilon > 0$, it then follows that

$$\sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{2} \left(-\pi_{ij} + \frac{R_i}{\rho_{ij}}\right)(\pi_{ij} - \pi_m)$$

$$= \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{2} \left[-(\pi_{ij} - \pi_m)^2 + \left(\frac{R_i}{\rho_{ij}} - \pi_m\right)(\pi_{ij} - \pi_m)\right]$$

$$\le \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{2} \left[-\frac{1}{2}(\pi_{ij} - \pi_m)^2 + \frac{1}{2}\left(\frac{R_i}{\rho_{ij}} - \pi_m\right)^2\right].$$

Thus, selecting a π_m such that $\pi_m \geq \bar{\beta}\sqrt{p}$ yields that

$$\dot{V} \leq -X^T Q X + \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4}\left(\frac{R_i}{\rho_{ij}} - \pi_m\right)^2$$

$$+ \frac{\pi_m}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\eta e^{-ct} - \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4}(\pi_{ij} - \pi_m)^2$$

$$+ \frac{\bar{\alpha}N_{\max}(N-1)}{2}\sum_{i=1}^{N}\|e_{x_i} + e_{w_i}\|_1$$

$$+ \sum_{i=1}^{N}(e_{x_i} + e_{w_i})^T \sum_{j=1}^{N} a_{ij}\pi_{ij}h(\hat{x}_i - \hat{x}_j + \hat{w}_i - \hat{w}_j, t),$$

where we have used the Hölder's inequality. Then implementing the triggering condition

(3.17)-(3.18) yields

$$\dot{V} \leq -X^T Q X + \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4}\left(\frac{R_i}{\rho_{ij}} - \pi_m\right)^2 + \sum_{i=1}^{N}\epsilon_i e^{-\varphi_i t}$$

$$+ \frac{\pi_m}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\eta e^{-ct} - \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4}(\pi_{ij} - \pi_m)^2$$

$$\leq -\lambda_{Q/P}V + \sum_{i=1}^{N}\epsilon_i e^{-\varphi_i t} + \frac{\pi_m}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\eta e^{-ct}$$

$$- \frac{1}{4}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\left(\rho_{ij} - \lambda_{Q/P}\right)(\pi_{ij} - \pi_m)^2$$

$$+ \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4}\left(\frac{R_{2i}}{\rho_{ij}} - \pi_m\right)^2$$

$$\leq -\lambda_{Q/P}V + \sum_{i=1}^{N}\epsilon_i e^{-\varphi_i t} + \frac{\pi_m}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\eta e^{-ct}$$

$$+ \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4}\left(\frac{R_i}{\rho_{ij}} - \pi_m\right)^2,$$

where $\lambda_{Q/P} = \frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)}$, and the last inequality holds because $\rho_{ij} > \lambda_{Q/P}$. According to the

Comparison Lemma in [62], it holds that

$$V \le e^{-\lambda_{Q/P}t} \left[V(0) + \lambda_{Q/P} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4} \left(\frac{R_i}{\rho_{ij}} - \pi_m \right)^2 \right]$$

$$+ \lambda_{Q/P} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4} \left(\frac{R_i}{\rho_{ij}} - \pi_m \right)^2$$

$$+ e^{-\lambda_{Q/P}t} \sum_{i=1}^{N} \int_0^t \left(\epsilon_i e^{-(\varphi_i - \lambda_{Q/P})\tau} + \frac{\pi_m}{2} |\mathcal{N}_i| \eta e^{-(c - \lambda_{Q/P})\tau} \right) d\tau.$$

Therefore, $\lim_{t\to\infty} V = \lambda_{Q/P} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{4} \left(\frac{R_i}{\rho_{ij}} - \pi_m \right)^2$, which implies that $\left\| x_i - \frac{1}{N} \sum_{j=1}^{N} x_j \right\|_2$,

$\left\| v_i - \frac{1}{N} \sum_{j=1}^{N} v_j \right\|_2$, and $\left\| w_i - \frac{1}{N} \sum_{j=1}^{N} w_j \right\|_2$ are all bounded.

Second, define S_x, S_v and S_w as in the proof of Theorem 38. Note that $\sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij}\pi_{ij}h(\hat{\chi}_i - \hat{\chi}_j) = 0$ holds for any $i \in \mathcal{V}$ because $\pi_{ij} = \pi_{ji}$ and $h(-z,t) = -h(z,t)$. As a result, by a similar proof of Theorem 38, it follows that $\lim_{t\to\infty} \sum_{j=1}^{N} x_j = \sum_{j=1}^{N} x_j^r$ and $\lim_{t\to\infty} \sum_{j=1}^{N} v_j = \sum_{j=1}^{N} v_j^r$. Therefore, $\lim_{t\to\infty} \left(x_i - \frac{1}{N} \sum_{j=1}^{N} x_j^r \right)$ and $\lim_{t\to\infty} \left(v_i - \frac{1}{N} \sum_{j=1}^{N} v_j^r \right)$ are bounded.

For the proof of the statement (ii), we consider the following Lyapunov function candidate as

$$V_2 = \frac{1}{2} X^T P X + \frac{1}{4} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\pi_{ij} - \frac{R_i}{\rho_{ij}} \right)^2.$$

Taking the derivative yields that

$$\dot{V}_2 \le -X^T Q X - (e_x + e_w)^T (M \otimes I_p)\alpha$$

$$+ \sum_{i=1}^{N} (e_{x_i} + e_{w_i})^T \sum_{j=1}^{N} a_{ij}\pi_{ij}h(\hat{x}_i - \hat{x}_j + \hat{w}_i - \hat{w}_j, t)$$

$$- \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij}\rho_{ij}}{2} \left(\pi_{ij} - \frac{R_i}{\rho_{ij}} \right)^2 + \frac{\beta}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \|\hat{\chi}_i - \hat{\chi}_j\|_1$$

$$- \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{R_i}{\rho_{ij}} a_{ij} (\hat{\chi}_i - \hat{\chi}_j)^T h(\hat{\chi}_i - \hat{\chi}_j, t).$$

Notice that $-(e_x + e_w)^T (M \otimes I_p)\alpha \le \beta \|e_x + e_w\|_1$ Implementing the triggering condition

56

(3.17)-(3.18) yields that

$$\dot{V}_2 \leq -X^T Q X + \sum_{i=1}^{N} \epsilon_i e^{-\varphi_i t} + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{R_i}{\rho_{ij}} a_{ij} \eta e^{-ct}$$

$$- \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{a_{ij} \rho_{ij}}{2} \left(\pi_{ij} - \frac{R_i}{\rho_{ij}} \right)^2$$

$$\leq -\lambda_{Q/P} V + \sum_{i=1}^{N} \epsilon_i e^{-\varphi_i t} + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{R_i}{\rho_{ij}} a_{ij} \eta e^{-ct}$$

$$- \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \left(\frac{\rho_{ij}}{2} - \frac{\lambda_{Q/P}}{4} \right) \left(\pi_{ij} - \frac{R_i}{\rho_{ij}} \right)^2$$

$$\leq -\lambda_{Q/P} V + \sum_{i=1}^{N} \epsilon_i e^{-\varphi_i t} + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{R_i}{\rho_{ij}} a_{ij} \eta e^{-ct},$$

where the last inequality holds by noting that $\rho_{ij} > \lambda_{Q/P}$ in the statement. Following the similar line of analysis as in the proof of statement (1), we have $\lim_{t \to \infty} V_2 = 0$, which implies that $x_i \to \frac{1}{N} \sum_{j=1}^{N} x_j$, $v_i \to \frac{1}{N} \sum_{j=1}^{N} v_j$, and $w_i \to \frac{1}{N} \sum_{j=1}^{N} w_j$, as $t \to \infty$. Hence, the consensus step is completed. The sum-tracking step can be completed by the same analysis to that in the proof of statement (i). Therefore, the distributed average tracking is achieved with zero tracking error.

Next we prove that the proposed event-triggering mechanism (3.17)-(3.18) is able to exclude Zeno behavior. Since V (or V_2) is bounded according to the analysis above, it is concluded that $\|x_i\|_1$, $\|w_i\|_1$ $\forall i \in \mathcal{V}$ and $|\pi_{ij}|$ $\forall (i,j) \in \mathcal{E}$ are all bounded. It then follows that $\|\dot{x}_i\|_1$, $\|\dot{w}_i\|_1$ are bounded. Let $\dot{w}_i^{\max} = \sup_{t \in [0,\infty)} \|\dot{x}_i\|_1$, and $\dot{w}_i^{\max} = \sup_{t \in [0,\infty)} \|\dot{w}_i\|_1$.

Note that

$$\left| \|e_{x_i} + e_{w_i}\|_1 R_i + (e_{x_i} + e_{w_i})^T \hat{\zeta}_i \right| \le \|e_{xi} + e_{wi}\|_1 R_i + \left| (e_{xi} + e_{wi})^T \hat{\zeta}_i \right|$$

$$\le \|e_{x_i} + e_{w_i}\|_1 \left(R_i + \left\| \hat{\zeta}_i \right\|_\infty \right)$$

$$\le \|\hat{x}_i - x_i + \hat{w}_i - w_i\|_1 \left(R_i + \left\| \hat{\zeta}_i \right\|_\infty \right)$$

$$\le (t - t^*) (\dot{x}_i^{\max} + \dot{w}_i^{\max}) \left(R_i + \left\| \hat{\zeta}_i \right\|_\infty \right),$$

where $\hat{\zeta}_i$ is defined in (3.18). The next event will not be triggered before $\left| \|e_{x_i} + e_{w_i}\|_1 R_i + (e_{x_i} + e_{w_i})^T \hat{\zeta}_i \right| = \epsilon_i e^{-\varphi_i t}$. Thus, a lower bound is given by $\tau^* = t - t^*$ that solves the equation

$$(t - t^*) (\dot{x}_i^{\max} + \dot{w}_i^{\max}) \left(R_i + \left\| \hat{\zeta}_i \right\|_\infty \right) \tau^* = \epsilon_1 e^{-\varphi_i \tau^*} e^{-\varphi_i t^*}.$$

It is apparent that $\tau^* > 0$, which implies no Zeno behavior. This completes the proof. ∎

From the triggering condition (3.17)-(3.18) and the proof of Theorem 40, the function $\epsilon_i e^{-\phi_i t}$ serves as the time-varying threshold for the term $\left| \|e_{x_i} + e_{w_i}\|_1 R_i + (e_{x_i} + e_{w_i})^T \hat{\zeta}_i \right| := F(e_{x_i} + e_{w_i})$. Once $F(e_{x_i} + e_{w_i})$ reaches the threshold, the agent is triggered. Therefore, selecting proper ϵ_i and ϕ_i allows one to affect the rate of triggering times. To be exact, a larger value of ϵ_i and a smaller value of ϕ_i intuitively lead to a lower triggering rate.

The matrices P and Q are accessible to agents since once κ is determined, the form of these two matrices are fixed. Then, the eigenvalues of P and Q can be easily computed by each agent.

Remark 29 *As indicated in Theorem 40, the lower bound of the design parameter R_i depends on some global information such as the total number of agents in the network and the bounds related to the reference signals. However, the parameter is constant and can be*

58

determined off-line before running the algorithms. One can always be more conservative to select a large enough number for R_i. Moreover, due to the challenging nature of the problem studied in this chapter, it might be inevitable to have certain piece of global information to determine the lower bound for the design parameter. This is also the case in the literature [45, 64], even when solving a simpler problem compared to the one studied in this chapter. In addition, to obtain a better estimate of the lower bound of the design parameter, one can use some existing algorithms in the literature [51, 109] to estimate the global information by interacting with local neighbors.

Remark 30 *The adaptation law (3.15) is partially inspired by [128]. The difference is the adoption of R_i in (3.15) for each agent. From Theorem 40, we can see that the value of R_i has an effect on the tracking error. As stated in Theorem 40, distributed average tracking is achieved with zero tracking error when R_i is sufficient larger. It is also worth noting that in this chapter, an event-triggered communication mechanism is proposed to avoid continuous interactions and reduce the communication cost. In addition, only position measurements are used. These two points distinguish the present work from the one in [128].*

Remark 31 *The distributed average tracking problem is solved by the proposed event-triggered algorithm, and Zeno behavior is excluded, which removes the requirement of continuous interactions among agents. Compared with the existing works on event-triggered distributed average tracking algorithms [45, 64], the proposed one (3.13)-(3.15) contributes in the following two aspects: i) the algorithm is able to be implemented for double-integrator agents without using velocity measurements, which is economical and energy efficient; ii) several practical limitations have been overcome by the newly designed triggering strategy.*

3.4 Illustrative Examples

In this section, we provide examples to illustrate the results obtained in this chapter.

We consider a group of twenty physical agents ($N = 20$) given in (3.1), which are labeled as $1, \ldots, 20$. The agents form a ring topology. In the simulation, we set $u_i^r = A_i \sin(\vartheta_i t + \varphi_i)$ in (3.2) with $A_i = -0.04(0.7i+0.5)^2[2(i-3.5)-2(-1)^i]$, $\vartheta_i = 0.2(0.7i+0.5)$, and $\varphi_i = (2i\pi/N) - \pi$.

Select $\kappa = 5$ and $\pi_{ij}(0) = 1000$ for any i and j that are connected. Implement the algorithm (3.3)-(3.5) for (3.1). The simulation results are shown in Fig. 3.1. It can be seen that all the agents' physical states, positions and velocities, are capable of tracking $\frac{1}{20} \sum_{j=1}^{20} x_j^r$ and $\frac{1}{20} \sum_{j=1}^{20} v_j^r$, respectively.
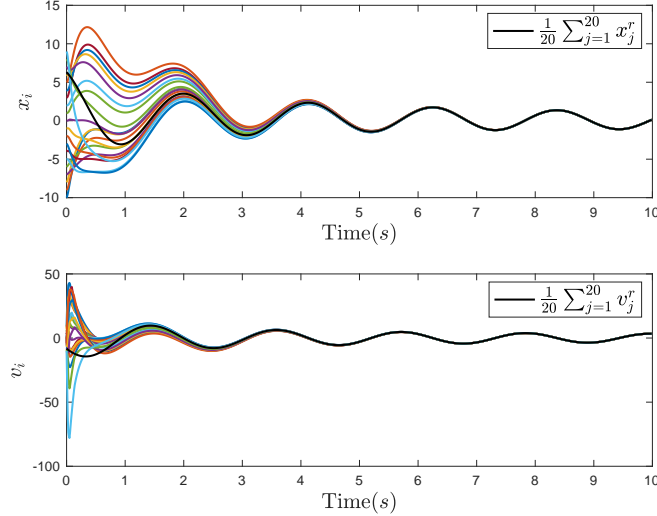

Figure 3.1: Using algorithm (3.3)-(3.5) for (3.1), twenty agents' position and velocity trajectories. The black lines denote the average of the reference signals and their velocities. The rest are the position and velocity trajectories of these twenty agents.

In the following, we use the algorithm (3.13)-(3.15) for (3.1) with the same set of reference signals. The triggering time instants are determined as in (3.17) with the triggering function defined in (3.18). For simplicity, we set $R_i = 2000$, $\epsilon_i = 1000$, $\rho_i = 5$ and $\varphi_i = 10^{-4}$ for any $i = 1, \ldots, 20$. Let $\eta = 10$ and $c = 1$. The position and velocity trajectories for those twenty agents are shown in Fig. 3.2. It can be seen that all the agents' physical states, positions and velocities, are capable of tracking $\frac{1}{20} \sum_{j=1}^{20} x_j^r$ and $\frac{1}{20} \sum_{j=1}^{20} v_j^r$, respectively. The number of triggering time instants for each agent is presented in Fig. 3.3. In this simulation, we use a fixed-step solver to solve the system, and the fixed-step size is 10^{-5}. In the 10 seconds simulation time, agents $1 - 20$ are triggered 3.69%, 3.80%, 3.84%, 3.93%, 3.91%, 4.03%, 3.99%, 4.08%, 4.02%, 3.90%, 4.09%, 3.78%, 4.02%, 3.79%, 3.85%, 3.68%, 3.49%, 3.27%, 3.34%, and 3.72% of times. Therefore, the proposed distributed average tracking algorithm (3.13)-(3.15) avoids continuous communication.
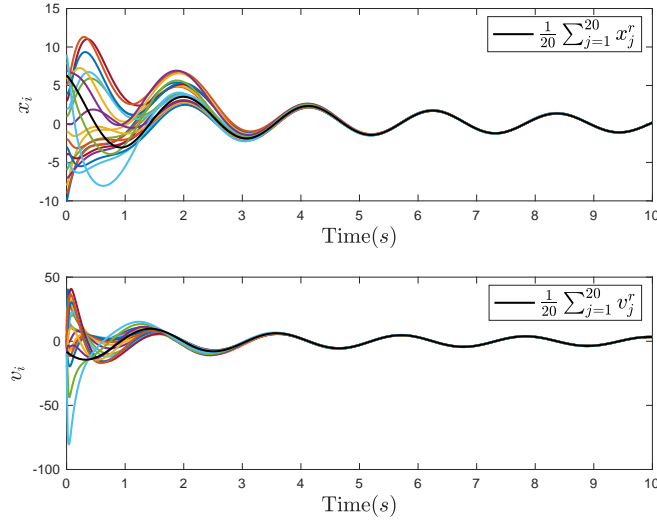

Figure 3.2: Using algorithm (3.13)-(3.15) for (3.1), twenty agents' position and velocity trajectories.

61
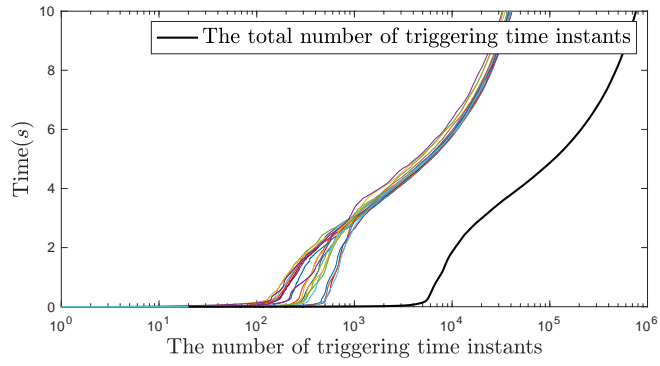
Figure 3.3: The number of triggering time instants of the agents while using algorithm (3.13)-(3.15) for (3.1). The black line denotes the total number of triggering time instants. The rest are the number of triggering time instants for these twenty agents.

Chapter 4

Distributed Time-varying Optimization of Networked Lagrangian Systems

4.1 Problem Statement

In the distributed time-varying optimization problem, each Lagrangian agent aims to cooperatively track the optimal trajectory determined by the group objective function. Let $q^*(t) \in \mathbb{R}^p$ denote the optimal trajectory, and it is defined as

$$q^*(t) = \arg \min_{q(t)} \left\{ \sum_{i=1}^{N} f_i[q(t), t] \right\}, \tag{4.1}$$

where $f_i[q(t), t] : \mathbb{R}^d \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ is the local cost function associated with agent $i \in \mathcal{V}$. In the rest of the chapter, it is assumed that $q^* \in \mathcal{L}_\infty^p$. This assumption is satisfied in most applications in practice. It is assumed that $f_i[q(t), t]$ is known only to agent i. Note that

$\sum_{i=1}^{N} f_i[q(t), t] = \sum_{i=1}^{N} f_i[q_i(t), t]$ if $q_i(t) = q_j(t) = q(t)$ for all $i, j \in \mathcal{V}$, and hence to find $q^*(t)$ defined in (4.1) is equivalent to find the optimal solution

$$\{q_1^*(t), \ldots, q_N^*(t)\} = \underset{\{q_1(t), \ldots, q_N(t)\}}{\arg\min} \left\{ \sum_{i=1}^{N} f_i[q_i(t), t] \right\},$$

$$\text{Subject to } q_i(t) = q_j(t) \quad \forall i \neq j,$$

where $q_i^*(t) = q_j^*(t) = q^*(t) \ \forall i \neq j$. Therefore, in this chapter, the goal is to design the control torques τ_i, $i \in \mathcal{V}$, for the agents (1.1) such that each agent's position $q_i(t)$ is capable of tracking $q_i^*(t) = q^*(t)$, and $\{q_1^*(t), \ldots, q_N^*(t)\}$ minimizes the group cost function $\sum_{i=1}^{N} f_i[q_i(t), t]$. That is, design τ_i for each agent i such that $\lim_{t \to \infty} [q_i(t) - q^*(t)] = \mathbf{0}_p$, $\forall i \in \mathcal{V}$. We make the following assumptions on the cost functions.

Assumption 4 *Each cost function $f_i(q_i, t)$, $i \in \mathcal{V}$, is twice continuously differentiable both in $q_i \in \mathbb{R}^p$ and t, and strongly convex in q_i and uniformly in t. That is, $H_i(q_i, t)$ is always positive definite and there exists a positive constant $\underline{m}$ such that $\lambda_j [H_i(q_i, t)] \geq \underline{m} \ \forall j \in \{1, \ldots, p\}$, $\forall i \in \mathcal{V}$ holds uniformly in t. In addition, each $H_i(q_i, t)$ is upper-bounded, i.e., $\|H_i(q_i, t)\|_2 \leq \bar{m} \ \forall i \in \mathcal{V}$.*

Assumption 5 *The Hessian matrices satisfy $H_i(q_i, t) = H_j(q_j, t) \ \forall i, j \in \mathcal{V}$.*

Assumption 6 *For each agent $i \in \mathcal{V}$, $\frac{\partial^2}{\partial t^2} \nabla f_i(q_i, t)$, $\frac{\partial^2}{\partial q_i^2} \nabla f_i(q_i, t)$ and $\frac{\partial^2}{\partial t \partial q_i} \nabla f_i$ exist. In addition, if agent i's position q_i, $i \in \mathcal{V}$, is bounded, then $\frac{\partial}{\partial t} \nabla f_i(q_i, t)$, $\frac{\partial^2}{\partial t^2} \nabla f_i(q_i, t)$, $\frac{\partial^2}{\partial q_i^2} \nabla f_i(q_i, t)$ and $\frac{\partial^2}{\partial t \partial q_i} \nabla f_i(q_i, t)$ are all bounded.*

In Assumption 4, the uniform strong convexity of the objective functions guarantees that the optimal trajectory q^* is unique for all $t \geq 0$, and it also ensures that $H_i(q_i, t)$

$\forall i \in \mathcal{V}$ is invertible for all t. The upper-boundedness of the Hessian matrix is equivalent to the Lipschitz continuity of the gradient $\nabla f_i(q_i, t)$. In Assumption 6, one sufficient condition for the existence of $\frac{\partial}{\partial t} \nabla f_i(q_i, t)$, $\frac{\partial^2}{\partial t^2} \nabla f_i(q_i, t)$, $\frac{\partial^2}{\partial q_i^2} \nabla f_i(q_i, t)$ and $\frac{\partial^2}{\partial t \partial q_i} \nabla f_i$, can be that each cost function $f_i(q_i, t)$, $i \in \mathcal{V}$, is at least three times continuously differentiable in q_i and t. Assumptions 4-6 are some similar/same assumptions that are used in prior related works [102, 92, 57].

Lemma 32 *[6] Let $f(x) : \mathbb{R}^p \to \mathbb{R}$ be a continuously differentiable convex function with respect to x. The function $f(x)$ is minimized at x^* if and only if $\nabla f(x^*) = \mathbf{0}_p$.*

4.2 Distributed Time-Varying Optimization of Networked Lagrangian Agents Under Fixed Graph

In this section, we assume the interaction topology of the Lagrangian agents is modeled as a fixed graph $\mathcal{G} \in \mathscr{G}$.

4.2.1 The Base Algorithm

For each agent $i \in \mathcal{V}$, construct a reference system as

$$
\begin{aligned}
\dot{v}_i = & -\sum_{j \in \mathcal{N}_i} \left[\alpha(q_i - q_j) + \beta(\dot{q}_i - \dot{q}_j) \right] \\
& - \gamma \sum_{j \in \mathcal{N}_i} \operatorname{sgn} \left[\alpha(q_i - q_j) + \beta(\dot{q}_i - \dot{q}_j) \right] + \varphi_i,
\end{aligned}
\tag{4.2}
$$

where α and β are some positive constants to be determined, and φ_i is defined by

$$
\varphi_i = -\dot{F}_i(q_i, t) - H_i(q_i, t) \nabla f_i(q_i, t),
\tag{4.3}
$$

with

$$F_i(q_i, t) = H_i^{-1}(q_i, t) \left[\frac{\partial}{\partial t} \nabla f_i(q_i, t) + \nabla f_i(q_i, t) \right]. \tag{4.4}$$

Note that Assumptions 4 and 6 guarrantee the existence of φ_i, $i \in \mathcal{V}$. Define

$$s_i = \dot{q}_i - v_i. \tag{4.5}$$

The adaptive controller for the Lagrangian system (1.1) is given by

$$\tau_i = -K_i s_i + Y_i(q_i, \dot{q}_i, v_i, \dot{v}_i)\hat{\vartheta}_i, \tag{4.6}$$

$$\dot{\hat{\vartheta}}_i = -\Gamma_i Y_i^T(q_i, \dot{q}_i, v_i, \dot{v}_i)s_i, \tag{4.7}$$

where K_i and Γ_i are symmetric positive definite matrices, and $\hat{\vartheta}_i$ is the estimate of ϑ_i. In the algorithm, the reference system (4.2) generate a desired reference velocity v_i for each agent i, and the adaptive controller (4.6)-(4.7) is used to drive each agent's velocity $\dot{q}_i$ to track its local v_i, and in the meantime, q_i to track the optimal trajectory.

Remark 33 *It is worth emphasizing that the algorithm (4.2)-(4.7) does not rely on exchange of virtual variables between neighbors. Especially, the reference system (4.2) is driven by agents' physical state information, i.e., q_i, $\dot{q}_i$, $q_i - q_j$ and $\dot{q}_i - \dot{q}_j$. Such design excludes the usage of communication channels, and can be implemented by onboard sensors. This feature distinguishes this algorithm from existing results on distributed optimization of networked Lagrangian systems, e.g., [127, 133, 132], where inter-agent communication is required. In addition, the algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.2) addresses the distributed time-varying optimization problem with zero optimum-tracking error, while the works [133, 132] are limited to distributed time-invariant optimization, and the work [127]*

only addresses a special case of time-varying cost functions with nonzero bounded optimum-tracking errors. It is also worth pointing out that such design results in the fact that the reference systems and agents' dynamics are highly coupled. The convergence analysis of such coupled systems is quite complex and challenging. However, in the literature, when distributed control problems are addressed for nonlinear systems, networked virtual systems are constructed completely independent of the agents' dynamics, which makes the analysis much easier and more straightforward compared with our convergence analysis later.

Assumption 7 *For any $i, j \in \mathcal{V}$, there exist positive constants c_1 and c_2 such that $\|\varphi_i - \varphi_j\|_1 \leq c_1(\|q_i - q_j\|_1 + \|\dot{q}_i - \dot{q}_j\|_1) + c_2$.*

Remark 34 *Assumptions 4-7 can be satisfied in many situations in practice. If the cost function are constructed as $f_i(q_i, t) = \|q_i(t) - r_i(t)\|_2^2$ where $q_i(t) \in \mathbb{R}^p$ and $r_i(t) \in \mathbb{R}^p$ are agent i's position and local reference signal, respectively, the distributed time-varying optimization algorithms can be applied to address the distributed average tracking of net-worked agents, which has found applications in region following formation control [13] and coordinated path planning [108]. Note that Assumption 4 holds trivially from the above construction of $f_i(q_i, t)$. Also, the boundedness assumptions of r_i, $\dot{r}_i$ and $\ddot{r}_i$ are commonly placed when dealing with the distributed average tracking of networked agents [50], and such boundedness assumptions implies that Assumptions 6 and 7 hold. In addition, when the cost functions have a slightly more general form as $f_i(q_i, t) = \|\rho q_i + g_i(t)\|_2^2$, where $\rho \in \mathbb{R}_+$ and $g_i(t)$ is a time-varying function, which is a commonly used cost function for energy mini-mization [57, 50], Assumptions 6 and 7 are satisfied under the boundedness assumption of $g_i(t)$, $\dot{g}_i(t)$ and $\ddot{g}_i(t)$. It is also worth pointing out that under Assumption 5, the value of the*

constants c_1 and c_2 in Assumption 7 depend mostly on the structure of the cost functions and their state-independent parts.

4.2.2 Convergence Analysis

Before moving on to the convergence analysis of the distributed optimization algorithm, an essential lemma is presented, which is used later.

Lemma 35 *Let* $z = \begin{bmatrix} z_1^T, \ldots, z_N^T \end{bmatrix}$ *and* $s = \begin{bmatrix} s_1^T, \ldots, s_N^T \end{bmatrix}$ *where* $z_i \in \mathbb{R}^p$ *and* $s_i \in \mathbb{R}^p$ $\forall i \in \mathcal{V}$. *Suppose that* $\gamma \in \mathbb{R}_+$. *It holds that*

$$- \gamma z^T (B \otimes I_p) \mathrm{sgn} \left[\left(B^T \otimes I_p \right) (z + \beta s) \right]$$

$$\leq -\gamma \left\| (B^T \otimes I_p) z \right\|_1 + 2\gamma\beta \left\| (B^T \otimes I_p) s \right\|_1,$$

where B *is the incidence matrix associated with the graph* $\mathcal{G}$.

Proof: Define $\mathcal{P} = \{1, \ldots, p\}$, and let $z_{i,k}$ and $s_{i,k}$ denote the k-th entry in vector z_i and s_i. It holds that

$$- \gamma z^T (B \otimes I_p) \mathrm{sgn} \left[\left(B^T \otimes I_p \right) (z + \beta s) \right]$$

$$= -\gamma \sum_{(i,j) \in \mathcal{E}} (z_i - z_j)^T \mathrm{sgn}[z_i - z_j + \beta(s_i - s_j)]$$

$$= -\gamma \sum_{(i,j) \in \mathcal{E}} \sum_{k \in \mathcal{P}} (z_{i,k} - z_{j,k}) \mathrm{sgn}[z_{i,k} - z_{j,k} + \beta(s_{i,k} - s_{j,k})]$$

$$= -\gamma \sum_{k \in \mathcal{P}} \sum_{(i,j) \in \mathcal{E}} \Lambda_{i,j}^k,$$

where $\Lambda_{i,j}^k = (z_{i,k} - z_{j,k}) \mathrm{sgn} \left[z_{i,k} - z_{j,k} + \beta(s_{i,k} - s_{j,k}) \right]$, and the equalities are obtained by using the definition of the signum function. For any $k \in \mathcal{P}$, define

$$\mathcal{E}_0^k = \left\{ (i, j) \in \mathcal{E} \; \middle| \; z_{i,k} - z_{j,k} + \beta(s_{i,k} - s_{j,k}) = 0 \right\}.$$

Note that $\Lambda_{i,j}^k = 0$ if $(i,j) \in \mathcal{E}_0^k$. It then holds that

$$-\gamma z^T(B \otimes I_p)\mathrm{sgn}\left[\left(B^T \otimes I_p\right)(z + \beta s)\right]$$

$$= -\gamma \sum_{k \in \mathcal{P}} \sum_{(i,j) \in \mathcal{E} \setminus \mathcal{E}_0^k} \Lambda_{i,j}^k.$$

For any $(i,j) \in \mathcal{E} \setminus \mathcal{E}_0^k$, it holds that

$$-\gamma \Lambda_{i,j}^k$$

$$= -\gamma \frac{(z_{i,k} - z_{j,k})^2 + \beta(z_{i,k} - z_{j,k})(s_{i,k} - s_{j,k})}{|z_{i,k} - z_{j,k} + \beta(s_{i,k} - s_{j,k})|}$$

$$= -\gamma|z_{i,k} - z_{j,k} + \beta(s_{i,k} - s_{j,k})|$$

$$\quad + \gamma\beta \frac{(s_{i,k} - s_{j,k})[z_{i,k} - z_{j,k} + \beta(s_{i,k} - s_{j,k})]}{|z_{i,k} - z_{j,k} + \beta(s_{i,k} - s_{j,k})|}$$

$$\leq -\gamma|z_{i,k} - z_{j,k} + \beta(s_{i,k} - s_{j,k})| + \gamma\beta|s_{i,k} - s_{j,k}|$$

$$\leq -\gamma\big||z_{i,k} - z_{j,k}| - \beta|s_{i,k} - s_{j,k}|\big| + \gamma\beta|s_{i,k} - s_{j,k}|,$$

where the last inequality follows from the Triangle Inequality. For any $k \in \mathcal{P}$, define

$$\mathcal{E}_+^k = \left\{(i,j) \in \mathcal{E} \; \middle| \; |z_{i,k} - z_{j,k}| \geq \beta|s_{i,k} - s_{j,k}|\right\},$$

$$\mathcal{E}_-^k = \left\{(i,j) \in \mathcal{E} \; \middle| \; |z_{i,k} - z_{j,k}| < \beta|s_{i,k} - s_{j,k}|\right\}.$$

Then, it follows that

$$-\gamma \sum_{(i,j)\in\mathcal{E}\backslash\mathcal{E}_0^k} \Lambda_{i,j}^k$$

$$\leq -\gamma \sum_{(i,j)\in\mathcal{E}_+^k\backslash\mathcal{E}_0^k} \left(|z_{i,k}-z_{j,k}| - 2\beta|s_{i,k}-s_{j,k}|\right)$$

$$+\gamma \sum_{(i,j)\in\mathcal{E}_-^k} |z_{i,k}-z_{j,k}|$$

$$\leq -\gamma \sum_{(i,j)\in\mathcal{E}} |z_{i,k}-z_{j,k}| + 2\gamma\beta \sum_{(i,j)\in\mathcal{E}_+^k\backslash\mathcal{E}_0^k} |s_{i,k}-s_{j,k}|$$

$$+\gamma \sum_{(i,j)\in\mathcal{E}_0^k} |z_{i,k}-z_{j,k}| + 2\gamma\beta \sum_{(i,j)\in\mathcal{E}_-^k} |s_{i,k}-s_{j,k}|$$

$$\leq -\gamma \sum_{(i,j)\in\mathcal{E}} |z_{i,k}-z_{j,k}| + 2\gamma\beta \sum_{(i,j)\in\mathcal{E}} |s_{i,k}-s_{j,k}|,$$

where the last inequality holds due to the definition of $\mathcal{E}_-^k$. Hence,

$$-\gamma z^T(B\otimes I_p)\mathrm{sgn}\left[\left(B^T\otimes I_p\right)(z+\beta s)\right]$$

$$\leq -\gamma \sum_{k\in\mathcal{P}} \sum_{(i,j)\in\mathcal{E}} \left(|z_{i,k}-z_{j,k}| - 2\beta|s_{i,k}-s_{j,k}|\right)$$

$$= -\gamma \sum_{(i,j)\in\mathcal{E}} \|z_i - z_j\|_1 + 2\gamma\beta \sum_{(i,j)\in\mathcal{E}} \|s_i - s_j\|_1$$

$$= -\gamma \left\|(B^T\otimes I_p)z\right\|_1 + 2\gamma\beta \left\|(B^T\otimes I_p)s\right\|_1.$$

This completes the proof. ∎

Using the definition of s_i in (4.5), the reference system (4.2) can be rewritten as

$$\dot{q}_i = v_i + s_i \tag{4.8}$$

$$\dot{v}_i = -\sum_{j\in\mathcal{N}_i} \left[\alpha(q_i - q_j) + \beta(v_i - v_j + s_i - s_j)\right]$$

$$-\gamma \sum_{j\in\mathcal{N}_i} \mathrm{sgn}\left[\alpha(q_i - q_j) + \beta(v_i - v_j + s_i - s_j)\right] + \varphi_i. \tag{4.9}$$

70

By the system reformulation, the reference system (4.2) (i.e., (4.8)-(4.9)) can be viewed as a group of networked perturbed double-integrators with disturbances s_i, $i \in \mathcal{V}$. The following proposition shows that the system (4.8)-(4.9) is input-to-state-like stable from the disturbances (i.e., s_i, $i \in \mathcal{V}$) to the optimum-tracking errors (i.e., $q_i(t) - q^*(t)$, $i \in \mathcal{V}$). That is, optimum-tracking errors are bounded and convergent to zero if the disturbances are bounded in a certain sense and convergent to zero.

Proposition 36 *Consider a group of N agents, and their interaction is described by the graph $\mathcal{G}$. Each agent's dynamics are given by (4.8)-(4.9). Suppose that Assumptions 1-7 hold. Let α and β be chosen such that $\alpha > \frac{2k}{\lambda_2(L)}$ and $\beta > \frac{3k+2\sqrt{k[\alpha\lambda_2(L)+2k]+4\alpha\lambda_2(L)-k}}{4\alpha\lambda_2(L)-k}\alpha$ with $k = c_1 p\lambda_N(L)(N-1)^2|\mathcal{E}|$, and γ be chosen such that $\gamma > c_2(N-1)^2|\mathcal{E}|$. Then, the following two statements hold.*

1. *If $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p$ $\forall i \in \mathcal{V}$, it holds that $q_i - q^* \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$.*

2. *If $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p$ and $s_i(t) \to \mathbf{0}_p$ $\forall i \in \mathcal{V}$ as $t \to \infty$, it holds that $q_i(t) \to q^*(t)$ $\forall i \in \mathcal{V}$ as $t \to \infty$.*

Proof: The proof of statements is divided into two steps: the coordination step and the optimum-tracking step. In the coordination step, it is proved that the coordination errors, $q_i - \frac{1}{N}\sum_{j=1}^N q_j$ and $v_i - \frac{1}{N}\sum_{j=1}^N v_j$, are bounded and convergent to zero if s_i $\forall i \in \mathcal{V}$ are bounded and convergent to zero, respectively. In the optimum-tracking step, it is proved that $\sum_{j=1}^N \nabla f_j(q_j, t) \in \mathcal{L}_\infty^p$ if $s_j \in \mathcal{L}_2^p$ $\forall i \in \mathcal{V}$, and $\sum_{j=1}^N \nabla f_j(q_j, t) \to \mathbf{0}_p$ as $t \to 0$ if $s_j \in \mathcal{L}_2^p$ and $s_i \to \mathbf{0}_p$ $\forall i \in \mathcal{V}$. Hence, the statements follow by combining these two steps.

First, consider the coordination step. Let $q = \begin{bmatrix} q_1^T, \ldots, q_N^T \end{bmatrix}^T$, $v = \begin{bmatrix} v_1^T, \ldots, v_N^T \end{bmatrix}^T$, $s = \begin{bmatrix} s_1^T, \ldots, s_N^T \end{bmatrix}^T$, and $\varphi = \begin{bmatrix} \varphi_1^T, \ldots, \varphi_N^T \end{bmatrix}^T$. Define $x = (M \otimes I_p)q$ and $y = (M \otimes I_p)v$,

71

where $M = I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T$. Then it holds that

$$\dot{x} = y + (M \otimes I_p)s \tag{4.10}$$

$$\dot{y} = -(L \otimes I_p)(\alpha x + \beta y + \beta s) + (M \otimes I_p)\varphi$$

$$- \gamma(B \otimes I_p)\mathrm{sgn}[(B^T \otimes I_p)(\alpha x + \beta y + \beta s)]. \tag{4.11}$$

Define the function

$$V = \frac{1}{2}\begin{bmatrix} x \\ y \end{bmatrix}^T \left(\begin{bmatrix} 2\alpha\beta L & \alpha I_N \\ \alpha I_N & \beta I_N \end{bmatrix} \otimes I_p \right) \begin{bmatrix} x \\ y \end{bmatrix}. \tag{4.12}$$

Note that the function V is positive definite if $\frac{\alpha}{\beta^2} < 2\lambda_2(L)$. Taking the derivative along

the solution of (4.10)-(4.11) yields $\dot{V} = U_1 + U_2$ where

$$U_1 = -\alpha^2 x^T(L \otimes I_p)x - y^T[(\beta^2 L - \alpha I_N) \otimes I_p]y$$

$$+ 2\alpha\beta x^T(L \otimes I_p)s - \beta^2 y^T(L \otimes I_p)s + \alpha y^T(M \otimes I_p)s,$$

and

$$U_2 = (\alpha x^T + \beta y^T)(M \otimes I_p)\varphi - \gamma(\alpha x^T + \beta y^T)$$

$$\times (B \otimes I_p)\mathrm{sgn}\left[(B^T \otimes I_p)(\alpha x + \beta y + \beta s)\right].$$

Consider the term U_1. For notational simplicity, let $z = \alpha x + \beta y$ and $\xi = \left[x^T, y^T\right]^T$.
Note that

$$x^T(L \otimes I_p)s \leq \|x\|_2 \|L \otimes I_p\|_2 \|s\|_2$$

$$\leq \lambda_N(L)\sqrt{Np} \|x\|_1 \|s\|_\infty.$$

Similarly,

$$-y^T(L \otimes I_p)s \le \lambda_N(L)\sqrt{Np}\,\|y\|_1\,\|s\|_\infty,$$

$$y^T(M \otimes I_p)s = y^T s \le \|y\|_1\,\|s\|_\infty.$$

Then it holds that

$$U_1 \le -\alpha^2 \lambda_2(L)\,\|x\|_2^2 - \left[\beta^2 \lambda_2(L) - \alpha\right]\|y\|_2^2$$

$$- \beta^2 y^T(L \otimes I_p)s + 2\alpha\beta x^T(L \otimes I_p)s + \alpha y^T(M \otimes I_p)s$$

$$\le -X^T Q_1 X + 2\alpha\beta\lambda_N(L)\sqrt{Np}\,\|x\|_1\,\|s\|_\infty$$

$$+ \left[\beta^2 \lambda_N(L)\sqrt{Np} + \alpha\right]\|y\|_1\,\|s\|_\infty$$

$$\le -X^T Q_1 X + c_M(\|x\|_1 + \|y\|_1)\,\|s\|_\infty$$

$$\le -X^T Q_1 X + c_M\sqrt{2Np}\,\|\xi\|_2\,\|s\|_\infty,$$

where $X = \left[\,\|x\|_2, \|y\|_2\,\right]^T$, $Q_1 = \mathrm{diag}\{\alpha^2 \lambda_2(L), \beta^2 \lambda_2(L) - \alpha\}$, and $c_M = \max\left\{2\alpha\beta\lambda_N(L)\sqrt{Np}, \beta^2\lambda_N(L)\sqrt{Np}\right.$

$\left.\alpha\right\}$.

Consider the term U_2. Note that

$$\|z\|_1 = \sum_{i=1}^{N}\left\|\sum_{j=1,j\neq i}^{N}\frac{1}{N}(z_i - z_j)\right\|_1$$

$$\le \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1,j\neq i}^{N}\|z_i - z_j\|_1$$

$$\le \max_{i\in\mathcal{V}}\left\{\sum_{j=1,j\neq i}^{N}\|z_i - z_j\|_1\right\}$$

$$\le \frac{(N-1)}{2}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i}\|z_i - z_j\|_1$$

$$= (N-1)\left\|(B^T \otimes I_p)z\right\|_1,$$

and it follows from Assumption 7 that

$$\|(M \otimes I_p)\varphi\|_\infty \leq \|(M \otimes I_p)\varphi\|_1$$

$$\leq \frac{(N-1)}{2} \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i} \|\varphi_i - \varphi_j\|_1$$

$$\leq (N-1)\Big[c_1 \left\|(B^T \otimes I_p)x\right\|_1 + c_1 \left\|(B^T \otimes I_p)y\right\|_1$$

$$+ c_1 \left\|(B^T \otimes I_p)s\right\|_1 + c_2|\mathcal{E}|\Big].$$

Note that

$$\left\|(B^T \otimes I_p)z\right\|_1 \leq \alpha \left\|(B^T \otimes I_p)x\right\|_1 + \beta \left\|(B^T \otimes I_p)y\right\|_1,$$

$$\left\|(B^T \otimes I_p)x\right\|_1 \leq \sqrt{|\mathcal{E}|p} \left\|(B^T \otimes I_p)x\right\|_2 \leq k_1 \left\|x\right\|_2,$$

$$\left\|(B^T \otimes I_p)y\right\|_1 \leq k_1 \left\|y\right\|_2,$$

$$\left\|(B^T \otimes I_p)s\right\|_1 \leq k_1 \left\|s\right\|_2 \leq k_2 \left\|s\right\|_\infty$$

$$\left\|(B^T \otimes I_p)z\right\|_1 \leq k_1 \left\|z\right\|_2 \leq k_3 \left\|\xi\right\|_2,$$

where $k_1 = \sqrt{|\mathcal{E}|p\lambda_N(L)}$, $k_2 = \sqrt{N|\mathcal{E}|p^2\lambda_N(L)}$ and $k_3 = \sqrt{|\mathcal{E}|p\lambda_N(L)(\alpha^2 + \beta^2)}$. Then,

$$z^T(M \otimes I_p)\varphi \leq \|z\|_1 \|(M \otimes I_p)\varphi\|_\infty$$

$$\leq X^T Q_2 X + k_4 \left\|\xi\right\|_2 \left\|s\right\|_\infty + \pi \left\|(B^T \otimes I_p)z\right\|_1,$$

where $Q_2 = k \begin{bmatrix} \alpha & \frac{\alpha+\beta}{2} \\ \frac{\alpha+\beta}{2} & \beta \end{bmatrix}$, $k = c_1 k_1^2 (N-1)^2$, $k_4 = c_1(N-1)^2 Npk_3 \left\|B^T \otimes I_p\right\|$, and $\pi = c_2(N-1)^2|\mathcal{E}|$. From Lemma 35, it follows that

$$-\gamma z^T(B \otimes I_p)\mathrm{sgn}\left[(B^T \otimes I_p)(z + \beta s)\right]$$

$$\leq -\gamma \left\|(B^T \otimes I_p)z\right\|_1 + 2\gamma\beta \left\|(B^T \otimes I_p)s\right\|_1.$$

Then, it holds that

$$U_2 \leq -(\gamma - \pi) \left\| \left(B^T \otimes I_p \right) z \right\|_1 + 2\gamma\beta \left\| \left(B^T \otimes I_p \right) s \right\|_1$$

$$+ X^T Q_2 X + k_4 \left\| \xi \right\|_2 \left\| s \right\|_\infty$$

$$\leq -(\gamma - \pi) \sqrt{\lambda_2(L)} \left\| z \right\|_2 + 2\gamma\beta k_2 \left\| s \right\|_\infty$$

$$+ X^T Q_2 X + k_4 \left\| \xi \right\|_2 \left\| s \right\|_\infty .$$

Hence,

$$\dot{V} \leq -X^T Q X + \left(c_M \sqrt{2Np} + k_4 \right) \left\| \xi \right\|_2 \left\| s \right\|_\infty$$

$$- (\gamma - \pi) \sqrt{\lambda_2(L)} \left\| z \right\|_2 + 2\gamma\beta k_2 \left\| s \right\|_\infty ,$$

where $Q = Q_1 - Q_2$. Note that Q is positive definite if $\alpha > \frac{2k}{\lambda_2(L)}$ and $\beta > \frac{3k + 2\sqrt{k[\alpha\lambda_2(L) + 2k] + 4\alpha\lambda_2(L) - k}}{4\alpha\lambda_2(L) - k}\alpha$.

Then, $-X^T Q X \leq -\lambda_m \left\| X \right\|_2^2$, where λ_m is the smallest eigenvalue of Q, i.e., $\lambda_m = \lambda_1(Q)$.

$$\dot{V} \leq -\lambda_m \left\| \xi \right\|_2^2 + \left(c_M \sqrt{2Np} + k_4 \right) \left\| \xi \right\|_2 \left\| s \right\|_\infty$$

$$- (\gamma - \pi) \sqrt{\lambda_2(L)} \left\| z \right\|_2 + 2\gamma\beta k_2 \left\| s \right\|_\infty$$

$$= -\lambda_m(1 - 2\eta) \left\| \xi \right\|_2^2 - (\gamma - \pi) \sqrt{\lambda_2(L)} \left\| z \right\|_2$$

$$- 2\lambda_m \eta \left\| \xi \right\|_2^2 + \left(c_M \sqrt{2Np} + k_4 \right) \left\| \xi \right\|_2 \left\| s \right\|_\infty$$

$$+ 2\gamma\beta k_2 \left\| s \right\|_\infty ,$$

$\eta \in \left(0, \frac{1}{2} \right)$. Note that the term $-2\lambda_m \eta \left\| \xi \right\|_2^2 + \left(c_M \sqrt{2Np} + k_4 \right) \left\| \xi \right\|_2 \left\| s \right\|_\infty + 2\gamma\beta k_2 \left\| s \right\|_\infty$ is

non-positive if $\left\| \xi \right\|_2 \geq \max \left\{ d_1 \left\| s \right\|_\infty , d_2 \sqrt{\left\| s \right\|_\infty} \right\}$, where $d_1 = \frac{c_M \sqrt{2Np} + k_4}{\lambda_m \eta}$ and $d_2 = \sqrt{\frac{2\gamma\beta k_2}{\lambda_m \eta}}$.

Note that $\rho(r) = \max \left\{ d_1 r, d_2 \sqrt{r} \right\}$ is a class $\mathcal{K}$ function. It holds that

$$\dot{V} \leq -\lambda_m(1 - 2\eta) \left\| \xi \right\|_2^2$$

$$- (\gamma - \pi) \sqrt{\lambda_2(L)} \left\| z \right\|_2 \quad \forall \left\| \xi \right\|_2 \geq \rho(\left\| s \right\|_\infty).$$

It then follows from [62, Theorem 4.19] and the property of the input-to-state stability [62, p. 175] that $x \in \mathcal{L}_\infty^{Np}$ and $y \in \mathcal{L}_\infty^{Np}$ if $s \in \mathcal{L}_\infty^{Np}$, and that $x(t) \to \mathbf{0}_{Np}$ and $y(t) \to \mathbf{0}_{Np}$ as $t \to \infty$ if $s(t) \to \mathbf{0}_{Np}$ as $t \to \infty$. Then, the coordination step is concluded from the definitions of x and y.

Consider the optimum-tracking step. Let $\chi = \sum_{j=1}^{N} \nabla f_j(q_j, t)$ and $\psi = \sum_{j=1}^{N} \left[v_j + F_j(q_j, t) \right]$. It holds that

$$\dot{\chi} = -\chi + \sum_{j=1}^{N} H_j(q_j, t)(v_j + F_j) + \sum_{j=1}^{N} H_j(q_j, t)s_j \tag{4.13}$$

$$\dot{\psi} = -\sum_{j=1}^{N} H_j(q_j, t)\nabla f_j(q_j, t). \tag{4.14}$$

Define the Lyapunov function candidate

$$W = \frac{1}{2}\chi^T \chi + \frac{1}{2}\psi^T \psi. \tag{4.15}$$

Taking the derivative of W along the solution of the networked system (4.8)-(4.9) yields that

$$\dot{W} = \chi^T \left[\sum_{j=1}^{N} H_j(q_j, t)v_j + \sum_{j=1}^{N} \frac{\partial}{\partial t}\nabla f_j(q_j, t) \right]$$
$$+ \chi^T \left[\sum_{j=1}^{N} H_j(q_j, t)s_j \right] - \psi^T \left[\sum_{j=1}^{N} H_j(q_j, t)\nabla f_j(q_j, t) \right]^T$$
$$= -\chi^T \chi + \chi^T \left[\sum_{j=1}^{N} H_j(q_j, t)s_j \right],$$

where Assumption 5 has been used. Note that

$$\chi^T \left[\sum_{j=1}^{N} H_j(q_j, t)s_j \right] \leq \|\chi\|_2 \sum_{j=1}^{N} \bar{m} \|s_j\|_2$$
$$\leq \frac{1}{2} \|\chi\|_2^2 + \frac{N\bar{m}^2}{2} \sum_{j=1}^{N} \|s_j\|_2^2.$$

76

It follows that

$$\dot{W} \leq -\frac{1}{2}\|\chi\|_2^2 + \frac{N\bar{m}^2}{2}\sum_{j=1}^{N}\|s_j\|_2^2.$$

Then, it holds that $2\dot{W} + \|\chi\|_2^2 \leq N\bar{m}^2\sum_{j=1}^{N}\|s_j\|_2^2$. Integrating over $[0,t]$ on both sides yields that

$$2\int_0^t \dot{W}\,\mathrm{d}\tau + \int_0^t \|\chi\|_2^2\,\mathrm{d}\tau \leq N\bar{m}^2\sum_{j=1}^{N}\int_0^t \|s_j\|_2^2\,\mathrm{d}\tau,$$

which is equivalent to

$$2W(t) + \int_0^t \|\chi\|_2^2\,\mathrm{d}\tau \leq 2W(0) + N\bar{m}^2\sum_{j=1}^{N}\int_0^t \|s_j\|_2^2\,\mathrm{d}\tau.$$

If $s_j \in \mathcal{L}_2^p$ $\forall i \in \mathcal{V}$, it holds that $\int_0^t \|s_j\|_2^2\,\mathrm{d}\tau < \infty$ $\forall i \in \mathcal{V}$ $\forall t \geq 0$. Note that $W(t) \geq 0$ $\forall t \geq 0$. It then holds that $2W(t) + \int_0^t \|\chi\|_2^2\,\mathrm{d}\tau < \infty$ $\forall t \geq 0$, which implies that $W(t) \in \mathcal{L}_\infty^1$ and $\chi \in \mathcal{L}_2^p$. Hence, it follows from (4.15) that $\chi \in \mathcal{L}_\infty^p$ and $\psi \in \mathcal{L}_\infty^p$. By Assumption 4, it holds that function $\sum_{j=1}^{N} f_j(q,t)$ is strongly convex in q. Then it follows from Assumption 5 that $N\underline{m}\|\bar{q} - q^*\|_2^2 \leq \left[\sum_{j=1}^{N}\nabla f_j(\bar{q},t) - \sum_{j=1}^{N}\nabla f_j(q^*,t)\right]^T (\bar{q} - q^*) = \left[\sum_{j=1}^{N}\nabla f_j(\bar{q},t) - \sum_{j=1}^{N}\nabla f_j(q_j,t)\right]^T (\bar{q} - q^*) + \left[\sum_{j=1}^{N}\nabla f_j(q_j,t) - \sum_{j=1}^{N}\nabla f_j(q^*,t)\right]^T (\bar{q} - q^*)$. Then, it holds that

$$N\underline{m}\|\bar{q} - q^*\|_2 \leq \left\|\sum_{j=1}^{N}\nabla f_j(\bar{q},t) - \sum_{j=1}^{N}\nabla f_j(q_j,t)\right\|_2$$
$$+ \left\|\sum_{j=1}^{N}\nabla f_j(q_j,t) - \sum_{j=1}^{N}\nabla f_j(q^*,t)\right\|_2.$$

From Assumption 4 and Mean Value theorem, there exist a positive constant M such that

$$\left\|\sum_{j=1}^{N}\nabla f_j(\bar{q},t) - \sum_{j=1}^{N}\nabla f_j(q_j,t)\right\|_2 \leq \sum_{j=1}^{N}\|\nabla f_j(\bar{q},t) - \nabla f_j(q_j,t)\|_2 \leq \sum_{j=1}^{N} M\|q_j - \bar{q}\|.$$

Since $\sum_{j=1}^{N} f_j(q^*,t) = \mathbf{0}_p$, and recall that $q_i - \frac{1}{N}\sum_{j=1}^{N} q_j \in \mathcal{L}_\infty^p$, then it can be shown

77

that $\|\bar{q} - q^*\|_2 < \infty$, and hence $q_i - q^* \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$. This concludes the first part of the statement.

Recall that $\chi \in \mathcal{L}_2^p$ and $\psi \in \mathcal{L}_\infty^p$. If $s_j \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$, it follows from (4.13) that $\dot{\chi} \in \mathcal{L}_\infty^p$, which implies that χ is uniformly continuous. Recall also that $\chi \in \mathcal{L}_2^p$, then it follows from Barbalat's Lemma [62, p. 323] that $\sum_{j=1}^N \nabla f_j(q_j, t) \to \mathbf{0}_p$ as $t \to \infty$. Recall from the coordination step that $x_i \to x_j$ and $v_i \to v_j$ $\forall i, j \in \mathcal{V}$ as $t \to \infty$ if $s_i \to \mathbf{0}_p$ $\forall i \in \mathcal{V}$ as $t \to \infty$. Hence, if $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p$ and $s_i \to \mathbf{0}_p$ $\forall i \in \mathcal{V}$, it follows from Lemma 32 that $q_i(t) \to q^*(t)$ $\forall i \in \mathcal{V}$ as $t \to \infty$. This concludes the second part of the statement. ∎

Proposition 37 *Suppose that Assumptions 1-7 hold. For the system (4.8)-(4.9), if $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p$ $\forall i \in \mathcal{V}$, then all $\varphi_i \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$.*

Proof: From the optimum-tracking step in the proof of Proposition 36, it holds that $q_i \in \mathcal{L}_\infty^p$ and $\psi \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$. By Assumptions 4 and 6, it holds that $\nabla f_i(q_i, t) \in \mathcal{L}_\infty^p$ and $\frac{\partial}{\partial t} \nabla f_i(q_i, t) \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$. Hence, $v_i \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$. Note also that $\frac{\partial^2}{\partial t^2} \nabla f_i(q_i, t)$, $\frac{\partial^2}{\partial q_i^2} \nabla f_i(q_i, t)$ and $\frac{\partial^2}{\partial t \partial q_i} \nabla f_i$ are all bounded by Assumption 6. Thus, it follows that $\varphi_i \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$. ∎

With Propositions 36 and 37 at hand, the convergence of the distributed optimization algorithm (4.6)-(4.7) with v_i and $\dot{v}_i$ given by the reference system (4.2) can be established by the following theorem.

Theorem 38 *Suppose that Assumptions 1-7 hold, and let α and β be chosen such that $\alpha > \frac{2k}{\lambda_2(L)}$ and $\beta > \frac{3k + 2\sqrt{k[\alpha \lambda_2(L) + 2k] + 4\alpha \lambda_2(L) - k}}{4\alpha \lambda_2(L) - k} \alpha$ with $k = c_1 p \lambda_N(L)(N-1)^2 |\mathcal{E}|$, and γ be chosen such that $\gamma > c_2(N-1)^2 |\mathcal{E}|$. Using the controller (4.6)-(4.7) with v_i and $\dot{v}_i$ given by*

the reference system (4.2) *for the networked Lagrangian system* (1.1) *solves the distributed time-varying optimization problem, that is,* $q_i(t) \to q^*(t) \; \forall i \in \mathcal{V} \; as \; t \to \infty.$

Proof: For any $i \in \mathcal{V}$, define Lyapunov function candidate $W_i = \frac{1}{2} s_i^T M_i(q_i) s_i + \frac{1}{2} \Delta \vartheta_i^T \Gamma_i^{-1} \Delta \vartheta_i$ with $\Delta \vartheta_i = \hat{\vartheta}_i - \vartheta_i$. By using Property 2, the derivative of W_i along the trajectories of the Lagrangian system (1.1) with the adaptive controller (4.6)-(4.7) can be written as $\dot{W}_i = -s_i^T K_i s_i \leq 0$. Hence, it holds that $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p$ and $\hat{\vartheta}_i \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$.

Using (4.5), we can rewrite (4.2) as the system (4.8)-(4.9). Since $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p$ $\forall i \in \mathcal{V}$, it follows the analysis of Proposition 37 that $\varphi_i \in \mathcal{L}_\infty^p$, $q_i \in \mathcal{L}_\infty^p$ and $v_i \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$. From (4.8), it then holds that $\dot{q}_i \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$. From (4.9), it holds that $\dot{v}_i \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$.

Substituting (4.6) into (1.1) and using Property 3 yield that

$$M_i(q_i)\dot{s}_i + C_i(q_i, \dot{q}_i)s_i = -K_i s_i + Y_i(q_i, \dot{q}_i, v_i, \dot{v}_i)\Delta \vartheta_i. \tag{4.16}$$

Then by using Property 1 and (4.16), it holds that $\dot{s}_i \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$. It can thus be shown that $s_i \; \forall i \in \mathcal{V}$ are uniformly continuous. Recall that $s_i \in \mathcal{L}_2^p \; \forall i \in \mathcal{V}$. Using Barbalat's lemma [62, p. 323], we obtain that $s_i(t) \to \mathbf{0}_p$ as $t \to \infty$ for any i in $\mathcal{V}$. Then, from the second statement of Proposition 36, it follows that $q_i(t) \to q^*(t) \; \forall i \in \mathcal{V}$ as $t \to \infty$. ∎

4.2.3 Distributed Time-varying Optimization Algorithm Removing Chattering

Note that the control torques may involve the chattering issue in practice [103], since $\dot{v}_i$, given in (4.2), introduces the signum function in the controller (4.6). This subsection focuses on the distributed time-varying optimization algorithm that generates continuous control inputs, and hence removes the chattering issue while application in practice.

To reduce the effects of chattering, we introduce a differentiable function $h(\cdot)$ to approximate and replace the signum function, which results in continuous control torques for the Lagrangian agents. The function $h(\cdot)$ is given by

$$h(r) = \frac{r}{\|r\|_2 + \varepsilon} \tag{4.17}$$

where $r \in \mathbb{R}^p$ and ε is a positive constant. Such an idea of using continuous approximation can be found in [83]. Compared with the traditional stabilization of a single agent considered in [83], the time-varying optimization of networked nonlinear systems addressed here are more challenging, and theoretical proof can not be directly implied. After replacing the signum function in (4.2) with the function (4.17), the reference system for agent $i \in \mathcal{V}$ becomes

$$\dot{v}_i = -\sum_{j \in \mathcal{N}_i} \left[\alpha(q_i - q_j) + \beta(\dot{q}_i - \dot{q}_j) \right]$$
$$-\gamma \sum_{j \in \mathcal{N}_i} h\left[\alpha(q_i - q_j) + \beta(\dot{q}_i - \dot{q}_j) \right] + \varphi_i, \tag{4.18}$$

where φ_i is defined in (4.3). As in Section 4.2.2, a similar preliminary lemma is presented first, and it is then used in the convergence analysis.

Lemma 39 *Let $\gamma \in \mathbb{R}_+$, and $z_i, s_i \in \mathbb{R}^p$, $i \in \mathcal{V}$. It holds that*

$$-\gamma \sum_{(i,j) \in \mathcal{E}} (z_i - z_j)^T h\left[z_i - z_j + \beta(s_i - s_j) \right]$$
$$\leq -\gamma \sum_{(i,j) \in \mathcal{E}} \left(\|z_i - z_j\|_2 - 2\beta \|s_i - s_j\|_2 - \varepsilon \right), \tag{4.19}$$

where $\mathcal{E}$ is the edge set of the graph $\mathcal{G}$.

Proof: From the definition of $h(\cdot)$ in (4.17), it follows that

$$-\gamma \sum_{(i,j)\in\mathcal{E}} (z_i - z_j)^T h\big[z_i - z_j + \beta(s_i - s_j)\big]$$

$$= -\gamma \sum_{(i,j)\in\mathcal{E}} \|z_i - z_j + \beta(s_i - s_j)\|_2 + \gamma \sum_{(i,j)\in\mathcal{E}} \varepsilon$$

$$+ \gamma \sum_{(i,j)\in\mathcal{E}} \frac{\beta(s_i - s_j)^T\big[z_i - z_j + \beta(s_i - s_j)\big] - \varepsilon^2}{\|z_i - z_j + \beta(s_i - s_j)\|_2 + \varepsilon}.$$

By using the Cauchy–Schwarz inequality, it holds that $(s_i - s_j)^T\big[z_i - z_j + \beta(s_i - s_j)\big] \leq$

$\|s_i - s_j\|_2 \|z_i - z_j + \beta(s_i - s_j)\|_2$. It then follows that

$$-\gamma \sum_{(i,j)\in\mathcal{E}} (z_i - z_j)^T h\big[z_i - z_j + \beta(s_i - s_j)\big]$$

$$\leq -\gamma \sum_{(i,j)\in\mathcal{E}} \|z_i - z_j + \beta(s_i - s_j)\|_2 + \gamma \sum_{(i,j)\in\mathcal{E}} \beta \|s_i - s_j\|_2$$

$$+ \gamma \sum_{(i,j)\in\mathcal{E}} \varepsilon - \gamma \sum_{(i,j)\in\mathcal{E}} \frac{\beta\varepsilon \|s_i - s_j\|_2 + \varepsilon^2}{\|z_i - z_j + \beta(s_i - s_j)\|_2 + \varepsilon}.$$

Define

$$\mathcal{E}_+ = \{(i,j) \in \mathcal{E} \mid \|z_i - z_j\|_2 \geq \beta \|s_i - s_j\|_2\},$$

$$\mathcal{E}_- = \{(i,j) \in \mathcal{E} \mid \|z_i - z_j\|_2 < \beta \|s_i - s_j\|_2\}.$$

81

Since $\frac{\beta\varepsilon\|s_i - s_j\|_2 + \varepsilon^2}{\|z_i - z_j + \beta(s_i - s_j)\|_2 + \varepsilon} \geq 0 \ \forall (i,j) \in \mathcal{E}$, it then holds that

$$-\gamma \sum_{(i,j)\in\mathcal{E}} (z_i - z_j)^T h\big[z_i - z_j + \beta(s_i - s_j)\big]$$

$$\leq -\gamma \sum_{(i,j)\in\mathcal{E}} \big| \|z_i - z_j\|_2 - \beta\|s_i - s_j\|_2 \big|$$

$$+ \gamma \sum_{(i,j)\in\mathcal{E}} \big(\varepsilon + \beta\|s_i - s_j\|_2\big)$$

$$\leq -\gamma \sum_{(i,j)\in\mathcal{E}_+} \|z_i - z_j\|_2 + \gamma\beta \sum_{(i,j)\in\mathcal{E}_+} \|s_i - s_j\|_2$$

$$-\gamma\beta \sum_{(i,j)\in\mathcal{E}_-} \|s_i - s_j\|_2 + \gamma \sum_{(i,j)\in\mathcal{E}_-} \|z_i - z_j\|_2$$

$$+ \gamma \sum_{(i,j)\in\mathcal{E}} \big(\varepsilon + \beta\|s_i - s_j\|_2\big),$$

where the first inequality follows from the reverse triangle inequality. By the definition of $\mathcal{E}_-$, (4.19) follows. ∎

Theorem 40 *Suppose that Assumptions 1-7 hold, and let Let α and β be chosen such that $\alpha > \frac{2k}{\lambda_2(L)}$ and $\beta > \frac{3k + 2\sqrt{k[\alpha\lambda_2(L) + 2k] + 4\alpha\lambda_2(L) - k}}{4\alpha\lambda_2(L) - k}\alpha$ with $k = c_1 p\lambda_N(L)(N-1)^2|\mathcal{E}|$, and γ be chosen such that $\gamma > c_2(N-1)^2|\mathcal{E}|\sqrt{p}$. Using the controller (4.6)-(4.7) with $\dot{v}_i$ defined in (4.18) for the networked Lagrangian system (1.1) solves the distributed time-varying optimization problem with bounded optimum-tracking errors, that is, $q_i(t) - q^*(t) \in \mathcal{L}_\infty^p$ $\forall i \in \mathcal{V}$.*

Proof: We use the definition of s_i in (4.5) and rewrite (4.18) as

$$\dot{q}_i = v_i + s_i \tag{4.20}$$

$$\dot{v}_i = -\sum_{j \in \mathcal{N}_i} \left[\alpha(q_i - q_j) + \beta(v_i - v_j + s_i - s_j) \right] + \varphi_i$$

$$- \gamma \sum_{j \in \mathcal{N}_i} h\left[\alpha(q_i - q_j) + \beta(v_i - v_j + s_i - s_j) \right]. \tag{4.21}$$

We only need to prove a similar statement for the networked system (4.20)-(4.21) as in

Proposition 36: $q_i - q^* \in \mathcal{L}_\infty^p \ \forall i \in \mathcal{V}$ if $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p \ \forall i \in \mathcal{V}$. As in the proof of Proposition

36, we also divide the proof into the coordination step and the sum-tracking step. In the

coordination step, let q, v and φ denote the column stack vectors of all q_i's, v_i's and φ_i's,

$i \in \mathcal{V}$, respectively, and define $x = (M \otimes I_p)q$, $y = (M \otimes I_p)v$, and $z = \alpha x + \beta y$ as in the

proof of Proposition 36, then, we have

$$\dot{x} = y + (M \otimes I_p)s \tag{4.22}$$

$$\dot{y} = -(L \otimes I_p)(\alpha x + \beta y + \beta s) + (M \otimes I_p)\varphi - \gamma \mathcal{H}. \tag{4.23}$$

where $\mathcal{H} = \left[\mathcal{H}_1^T, \ldots, \mathcal{H}_N^T \right]$ with $\mathcal{H}_i = \sum_{j \in \mathcal{N}_i} h\left[z_i - z_j + \beta(s_i - s_j) \right]$. By using the same

Lyapunov function candidate as in the proof of Proposition 36, we have $\dot{V} = -X^T Q X +$

$\left(c_M \sqrt{2Np} + k_4 \right) \|\xi\|_2 \|s\|_\infty + \pi \left\| (B^T \otimes I_p)z \right\|_1 + 2\gamma\beta k_2 \|s\|_\infty - \gamma \sum_{(i,j) \in \mathcal{E}} (z_i - z_j)^T h\left[z_i - z_j + \right.$

$\beta(s_i - s_j)]$ where $X = \left[\|x\|_2, \|y\|_2 \right]^T$, $\xi = \left[x^T, y^T \right]^T$, $Q = \begin{bmatrix} \alpha^2 \lambda_2(L) - \alpha k & -\frac{\alpha+\beta}{2}k \\ -\frac{\alpha+\beta}{2}k & \beta^2 \lambda_2(L) - \alpha - \beta k \end{bmatrix}$,

$\pi = \frac{c_2(N-1)^3 N}{2}$, $k = c_1 k_1^2(N-1)^2$, $k_1 = \sqrt{|\mathcal{E}|p\lambda_N(L)}$, $k_2 = \sqrt{N|\mathcal{E}|p^2\lambda_N(L)}$, $k_3 =$

$\sqrt{|\mathcal{E}|p\lambda_N(L)(\alpha^2 + \beta^2)}$ and $k_4 = c_1(N-1)^2 Npk_3 \left\| B^T \otimes I_p \right\|_\infty$. By Lemma 39, it follows

that

$$-\gamma \sum_{(i,j)\in\mathcal{E}} (z_i - z_j)^T h\big[z_i - z_j + \beta(s_i - s_j)\big]$$

$$\leq -\gamma \sum_{(i,j)\in\mathcal{E}} \big(\|z_i - z_j\|_2 - 2\beta \|s_i - s_j\|_2 - \varepsilon \big)$$

$$\leq -\frac{\gamma}{\sqrt{p}} \left\| \big(B^T \otimes I_p\big) z \right\|_1 + k_5 \|s\|_1 + \gamma\varepsilon|\mathcal{E}|$$

where $k_5 = 2\gamma\beta \left\| B^T \otimes I_p \right\|_1$, and the second inequality is obtained by using the facts that

$\|r\|_2 \leq \|r\|_1 \leq \sqrt{p}\|r\|_2$ for any $r \in \mathbb{R}^p$. Hence,

$$\dot{V} \leq -\lambda_m \|\xi\|_2^2 + \big(c_M\sqrt{2Np} + k_4\big) \|\xi\|_2 \|s\|_\infty$$

$$- \left(\frac{\gamma}{\sqrt{p}} - \pi\right) \left\| \big(B^T \otimes I_p\big) z \right\|_1 + k_5 \|s\|_1 + \gamma\varepsilon|\mathcal{E}|$$

$$\leq -\lambda_m(1 - 3\eta) \|\xi\|_2^2 - \left(\frac{\gamma}{\sqrt{p}} - \pi\right)\sqrt{\lambda_2(L)} \|z\|_2$$

$$- 3\lambda_m\eta \|\xi\|_2^2 + \big(c_M\sqrt{2Np} + k_4\big) \|\xi\|_2 \|s\|_\infty$$

$$k_5 \|s\|_1 + \gamma\varepsilon|\mathcal{E}|,$$

where we have used the inequalities $-\left\| \big(B^T \otimes I_p\big) z \right\|_1 \leq -\left\| \big(B^T \otimes I_p\big) z \right\|_2 \leq -\sqrt{\lambda_2(L)}\|z\|_2$

and $\|s\|_\infty \leq \|\tilde{s}\|_\infty$ and the fact that $\gamma > \pi\sqrt{p}$ to obtain the second inequality. Note

that the term $-3\lambda_m\eta \|\xi\|_2^2 + \big(c_M\sqrt{2Np} + k_4\big) \|\xi\|_2 \|s\|_\infty k_5 \|s\|_1 + \gamma\varepsilon|\mathcal{E}|$ is non-positive if

$\|\xi\|_2 \geq \rho := \max\left\{ \frac{c_M\sqrt{2Np}+k_4}{\lambda_m\eta} \|s\|_\infty, \sqrt{\frac{k_5}{\lambda_m\eta}}\sqrt{\|s\|_\infty}, \sqrt{\frac{\gamma\varepsilon|\mathcal{E}|}{\lambda_m\eta}} \right\}$. It then holds that

$$\dot{V} \leq -\lambda_m(1 - 2\eta) \|\xi\|_2^2$$

$$- (\gamma - \pi)\sqrt{\lambda_2(L)} \|z\|_2 \quad \forall \|\xi\|_2 \geq \rho.$$

This shows that $\xi \in \mathcal{L}_\infty^{2Np}$, and hence $q_i - \frac{1}{N}\sum_{j=1}^{N} q_j \in \mathcal{L}_\infty^p$ and $v_i - \frac{1}{N}\sum_{j=1}^{N} v_j \in \mathcal{L}_\infty^p$.

The proof of the sum-tracking step is similar to that of Proposition 36 by noting that

84

$\sum_{i=1}^{N} \sum_{j \in \mathcal{N}_i(t)} h[\alpha(q_i - q_j) + \beta(v_i - v_j + s_i - s_j)] = \mathbf{0}_p$, and thus is omitted. Therefore, it can be concluded that $q_i - q^* \in \mathcal{L}_\infty^p$ if $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p \ \forall i \in \mathcal{V}$ for networked system (4.20)-(4.21). For each $i \in \mathcal{V}$, by using the same Lyapunov function W_i in the proof of Theorem 38, we have $s_i \in \mathcal{L}_\infty^p \cap \mathcal{L}_2^p \ \forall i \in \mathcal{V}$. Hence, we can conclude that $q_i - q^* \in \mathcal{L}_\infty^p$. That is, the distributed time-varying optimization problem of networked Lagrangian agents is solved with bounded optimum-tracking errors. ∎

There are other continuous functions that can be used to approximate the signum function, such as $\text{sat}\left(\frac{r}{\varepsilon}\right)$ and $\tanh\left(\frac{r}{\varepsilon}\right)$ where $\varepsilon \in \mathbb{R}_+$. When the signum function in (4.2) is replaced with a time-varying function $h(r) = \frac{r}{\|r\|_2 + \kappa_1 e^{-\kappa_2 t}}$ [34], it can be shown that $q_i(t) - q^*(t) \to \mathbf{0}_p \ \forall i \in \mathcal{V}$ as $t \to \infty$. The proof follows by the similar line of analysis in Lemma 39 and Theorem 40, which is omitted.

Remark 41 *The design of the reference system in* (4.2) *is inspired by the work [92], and the method of approximating the signum function using* (4.17) *has been applied in [92] to remove the cahttering. However, this work considers the distributed time-varying optimization problem for networked Lagrangian systems, and proposed algorithms can be implemented by using on-board sensors taking physical state measurements (absolute and/or relative position and velocity measurement). The Lagrangian dynamics are more complex compared with single- and double-integrator agents considered in [92]. Moreover, the complexities of the problem of interest and agents' dynamics pose challenges in the convergence analysis. For instance, as an intermediate step in the convergence analysis, two statements are established for the networked system* (4.8)-(4.9) *(or* (4.20)-(4.21) *in the case of using signum function approximation), which can be seen as networked second-order systems perturbed by distur-*

bances s_i, $i \in \mathcal{V}$. Hence they are different from the disturbance-free double-integrator model considered in [92], and there are significant technical challenges.

4.3 Distributed Time-varying Optimization Algorithm Under Switching Graphs

In this section, we focus on solving the distributed time-varying optimization problem under switching graphs when the cost functions satisfy Assumption 4 and the following assumption.

Assumption 8 *For any $i \in \mathcal{V}$, the gradient of the cost function $f_i(q_i, t)$ can be written as $\nabla f_i(q_i, t) = Hq_i + g_i(t)$, where $g_i(t)$ is a smooth time-varying function. In addition, There exist a positive constant $\bar{g}$ such that $\sup_{t \in [0,\infty)} \|g_i(t)\|_2 \leq \bar{g}$, $\sup_{t \in [0,\infty)} \|\dot{g}_i(t)\|_2 \leq \bar{g}$ and $\sup_{t \in [0,\infty)} \|\ddot{g}_i(t)\|_2 \leq \bar{g}$ $\forall i \in \mathcal{V}$.*

For each agent $i \in \mathcal{V}$, construct the reference system as

$$\dot{v}_i = -\alpha \dot{q}_i - \gamma \sum_{j \in \mathcal{N}_i(t)} \text{sgn}(q_i - q_j + \dot{q}_i - \dot{q}_j) + \varphi_i, \tag{4.24}$$

where α and γ are positive constants to be determined,

$$\varphi_i = -\alpha F_i(q_i, t) - \dot{F}_i(q_i, t), \tag{4.25}$$

and $F_i(q_i, t)$ is defined in (4.4). The adaptive controller for the Lagrangian system (1.1) is given by (4.6)-(4.7).

Theorem 42 *Suppose that Assumptions 1, 4 and 8 hold. Let $\alpha \in \mathbb{R}_+$ and $\gamma > \frac{2}{m}(\alpha + 1)(N-1)\bar{g}$. Using the controller (4.6)-(4.7) with $\dot{v}_i$ defined in (4.24) for the networked*

Lagrangian system (1.1) *solves the distributed time-varying optimization problem, that is,*

$q_i(t) \to q^*(t) \ \forall i \in \mathcal{V} \ as \ t \to \infty.$

Proof: From Assumption 8, it follows that $\varphi_i = -\alpha q_i - \dot{q}_i - H^{-1}\big[\alpha g_i(t) + (\alpha +$
$1)\dot{g}_i(t) + \ddot{g}_i(t)\big]$. Define $D_i = -H^{-1}\big[\alpha g_i(t) + (\alpha + 1)\dot{g}_i(t) + \ddot{g}_i(t)\big]$. Then, it follows from
Assumptions 4 and 8 that $\|D_i\|_\infty \le \frac{2}{m}(\alpha + 1)\bar{g} := \bar{D}$. It follows from (4.24) and (4.5) that

$$\dot{q}_i = v_i + s_i$$

$$\dot{v}_i = -\alpha q_i - (\alpha + 1)v_i - (\alpha + 1)s_i$$

$$- \gamma \sum_{j \in \mathcal{N}_i(t)} \mathrm{sgn}\big(q_i - q_j + v_i - v_j + s_i - s_j\big) + D_i.$$

Define $z_i = q_i + v_i$ and $z = [z_1^T, \ldots, z_N^T]^T$. It then holds that

$$\dot{z} = -\alpha z - \alpha s$$

$$- \gamma \big[B(t) \otimes I_p\big]\mathrm{sgn}\big\{\big[B^T(t) \otimes I_p\big](z + s)\big\} + D,$$

where $s = [s_1^T, \ldots, s_N^T]^T$ and $D = [D_1^T, \ldots, D_N^T]^T$. Define $x = (M \otimes I_p)z$. It holds that

$$\dot{x} = -\alpha x - \alpha(M \otimes I_p)s + (M \otimes I_p)D$$

$$- \gamma \big[B(t) \otimes I_p\big]\mathrm{sgn}\big\{\big[B^T(t) \otimes I_p\big](x + s)\big\}. \tag{4.26}$$

Consider the Lyapunov function candidate $V = \frac{1}{2}x^T x$. Taking the derivative of V yields
that

$$\dot{V} = -\alpha \|x\|_2^2 - \alpha x^T(M \otimes I_p)s + x^T(M \otimes I_p)D$$

$$- \gamma x^T\big[B(t) \otimes I_p\big]\mathrm{sgn}\big\{\big[B^T(t) \otimes I_p\big](x + s)\big\}.$$

Note that $x^T(M \otimes I_p)D \le (N-1)\left\|(B^T \otimes I_p)x\right\|_1 \|D\|_\infty \le \bar{D}(N-1)\left\|(B^T \otimes I_p)x\right\|_1$. By

Lemma 35, it holds that

$$- \gamma x^T\left[B(t) \otimes I_p\right]\mathrm{sgn}\left\{\left[B^T(t) \otimes I_p\right](x+s)\right\}$$

$$\le -\gamma\left\|\left[B^T(t) \otimes I_p\right]x\right\|_1 + 2\gamma\left\|\left[B^T(t) \otimes I_p\right]s\right\|_1$$

$$\le -\gamma\left\|\left[B^T(t) \otimes I_p\right]x\right\|_1 + k_1\|s\|_1$$

where $k_1 = \gamma N(N-1)$, and we have used the fact that $2\gamma\left\|\left[B^T(t) \otimes I_p\right]s\right\|_1 \le 2\gamma\left\|\left[B^T(t) \otimes I_p\right]\right\|_1 \|s\|_1 \le$

$\gamma N(N-1)\|s\|_1$ to obtain the second inequality. Then, it holds that

$$\dot{V} \le -\alpha\|x\|_2^2 + \alpha\|x\|_2\|s\|_1 - \gamma\left\|\left(B^T \otimes I_p\right)x\right\|_1$$

$$+ k_1\|s\|_1 + \bar{D}(N-1)\left\|(B^T \otimes I_p)x\right\|_1$$

$$\le -\alpha\|x\|_2^2 + \alpha\|x\|_2\|s\|_1 + k_1\|s\|_1$$

$$\le -\frac{\alpha}{2}\|x\|_2^2 + \frac{\alpha}{2}\|s\|_1^2 + k_1\|s\|_1,$$

where we have used the fact that $\gamma \ge \bar{D}(N-1)$ to obtain the second inequality and used

the Young's inequality $\|x\|_2\|s\|_1 \le \frac{1}{2}\|x\|_2^2 + \frac{1}{2}\|s\|_1^2$ to obtain the last inequality. Define

continuous functions $\rho(r) = \frac{\alpha}{2}r^2$ and $\varpi(r) = \frac{\alpha}{2}r^2 + k_1 r$, which are class $\mathcal{K}_\infty$ functions.

Then, it holds that

$$\dot{V} \le -\rho(\|x\|_2) + \varpi(\|s\|_1),$$

which implies that (V, ρ, ϖ) is a common ISS-Lyapunov triple[1] for the switched system

(4.26). By [77, Theorem 2.1], it holds that the system (4.26) is uniformly (with respect to the

---

[1]A common ISS-Lyapunov triple (V, ρ, ϖ) for the switched system $\dot{x} = f_\sigma(x, u)$ consists of a positive definite radially unbounded continuous differential function $V : \mathbb{R}^n \to [0, +\infty)$ and class $\mathcal{K}$ functions ρ and ϖ such that $\nabla V(x)f_\sigma(x, u) \le -\rho(\|x\|_2) + \varpi(\|u\|_2)$ for all $x \in \mathbb{R}^n$, all $u \in \mathbb{R}^m$ and all $\sigma \in \Gamma$, where Γ is the index set for the switched system [77].

switching signal for the graph) input-to-state stable. That is, there exist a class $\mathcal{KL}$ function ω and a class $\mathcal{K}$ function μ such that $\|x(t)\|_2 \leq \omega(\|x(0)\|_2, t) + \mu\left(\sup_{0\leq\tau\leq t}\|s(\tau)\|_1\right)$. Hence, it can be concluded that $x \in \mathcal{L}_\infty^{Np}$ if $s \in \mathcal{L}_\infty^{Np}$ and $x \to \mathbf{0}_{Np}$ if $s \to \mathbf{0}_{Np}$ as $t \to \infty$.

From the definitions of x and z, it holds that

$$(M \otimes I_p)\dot{q} = -(M \otimes I_p)q + (M \otimes I_p)s + x.$$

Note that $(M \otimes I_p)\dot{q} = -(M \otimes I_p)q$ is a standard exponentially stable linear time-invariant (LTI) system. Then, it holds that $q_i - \frac{1}{N}\sum_{j=1}^{N} q_j \in \mathcal{L}_\infty^p$ and $v_i - \frac{1}{N}\sum_{j=1}^{N} v_j \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$ if $s_i \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$ and $q_i \to \frac{1}{N}\sum_{j=1}^{N} q_j$ and $v_i \to \frac{1}{N}\sum_{j=1}^{N} v_j \; \forall i \in \mathcal{V}$ as $t \to \infty$ if $s_i \to \mathbf{0}_p$ as $t \to \infty$.

Use the same definitions of χ and ψ as in the proof of Proposition 36, and it holds that

$$\dot{\psi} = -\alpha\psi - \alpha\sum_{j=1}^{N} s_j. \tag{4.27}$$

Note that $\dot{\psi} = -\alpha\psi$ is a standard exponentially stable LTI system. Then, the system (4.27) is input-to-state stable. Hence, $\psi \in \mathcal{L}_\infty^p$ if $s_i \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$ and $\psi \to \mathbf{0}_d$ as $t \to \infty$ if $s_i \to \mathbf{0}_p$ $\forall i \in \mathcal{V}$ as $t \to \infty$.

Note that (4.13) holds. It then holds that $\chi \in \mathcal{L}_\infty^p$ if $s_i \in \mathcal{L}_\infty^p \; \forall i \in \mathcal{V}$ and $\chi \to \mathbf{0}_d$ as $t \to \infty$ if $s_i \to \mathbf{0}_p \; \forall i \in \mathcal{V}$ as $t \to \infty$. The rest of the proof follows from the same analysis in Theorem 38, which is thus omitted. ∎

Remark 43 *As shown in Remark 34, the distributed time-varying optimization algorithms can be used to solve the distributed average tracking of networked Lagrangian agents, which is the topic investigated in [16]. The proposed algorithms in this chapter has the following*

advantages over [16]. Firstly, zero tracking error can be guaranteed for general bounded reference signals, which cannot be done by using the algorithms in the work [16] in the same settings. Secondly, the algorithms only use absolute and/or relative measurements of the physical states with respect to the neighbors, while the work [16] requires addition communication of some virtual variable. Lastly, the algorithms (4.6)-(4.7) with $\dot{v}_i$ given by the reference system (4.24) works under switching graphs.

Remark 44 *The structure of the proposed distributed algorithms for networked Lagrangian agents are partially inspired by [115], where the consensus and/or leader-following of networked Lagrangian systems are investigated. In this chapter, the distributed time-varying optimization problem is addressed, which are more complex and challenging and include the consensus and leader-following as special cases. Moreover, while dealing with the distributed time-varying optimization for networked Lagrangian agents, the analysis is quite different from the work [115]. The nonlinear functions, such as the signum function and the one defined in (4.17), are used to constructing $\dot{v}_i$, which forms a perturbed closed-loop networked double-integrator systems with s_i as disturbance in the model and inside the nonlinear functions (see (4.8)-(4.9) for an example). This chapter provide rigorous analysis on the performance of the perturbed systems under bounded and convergent disturbances. In addition, when considering distributed time-varying optimization problem, additional analysis steps are required, see the optimum-tracking steps in the proof of Proposition 36 and Theorem 38 for instance.*

Remark 45 *As shown in Theorems 38-42, the lower bounds of the design parameters (e.g., γ, α, β and κ) depend on some global information, such as the bounds on the cost functions*

and the graph. It is worth mentioning that these design parameters are constants, and can be determined off-line. Once they are chosen, one can embed them into each agent and implement the proposed algorithms by using only local information (e.g., local cost functions, absolute/ relative measurements on physical states), which implies that the proposed algorithm can be implemented in a distributed way. In addition, one can be conservative and select large enough values for these parameters. Also, one can use some existing algorithms [109, 51] to estimate the bounds about the cost functions and the graph, and then choose appropriate values for the parameters based on the estimated bounds.

4.4 Illustrative Examples

In this section, we provide examples to illustrate the results in this chapter. We consider a group of ten planar manipulators with two revolute joints [105, pp. 259-262] ($N = 10$), which are labeled from 1 to 10 ($\mathcal{V} = \{1, \ldots, 10\}$). The interaction among these ten agents is characterized as the graph in Fig. 4.1. The i-th manipulator/agent's dynamics are given as

$$
\begin{cases}
d_{i11}\ddot{q}_{i1} + d_{i12}\ddot{q}_{i2} + c_{i11}\dot{q}_{i1} + c_{i12}\dot{q}_{i2} + g_{i1} = \tau_{i1} \\[2mm]
d_{i21}\ddot{q}_{i1} + d_{i22}\ddot{q}_{i2} + c_{i21}\dot{q}_{i1} + g_{i2} = \tau_{i2}
\end{cases}
$$

where $q_i = \begin{bmatrix} q_{i1}, q_{i2} \end{bmatrix}^T \in \mathbb{R}^2$ is the generalized coordinates, $d_{i11} = m_{i1}l_{ic1}^2 + m_{i2}(l_{i1}^2 + l_{ic2}^2 + 2l_{i1}l_{ic2}\cos q_{i2}) + J_{i1} + J_{i2}$, $d_{i12} = d_{i21} = m_{i2}(l_{ic2}^2 + l_{i1}l_{ic2}\cos q_{i2}) + J_{i2}$, $d_{i22} = m_{i2}l_{ic2}^2 + J_{i2}$, $c_{i11} = o\dot{q}_{i2}$, $c_{i12} = o(\dot{q}_{i2} + \dot{q}_{i1})$, $c_{i21} = -o\dot{q}_{i1}$, $o = -m_{i2}l_{i1}l_{ic2}\sin q_{i2}$, $g_{i1} = (m_{i1}l_{ic1} + m_{i2}l_{i1})g\cos q_{i1} + m_{i2}l_{ic2}g\cos(q_{i1} + q_{i2})$, $g_{i2} = m_{i2}l_{ic2}g\cos(q_{i1} + q_{i2})$, and g is the gravitational acceleration. For agent i, J_{ij}, m_{ij}, l_{ij}, and l_{icj} are the moment of inertia, mass, length, and

the distance from the previous joint to the center of mass of link j, respectively. Suppose that $\vartheta_i = [\vartheta_{i1}, \ldots, \vartheta_{i5}]$ where $\vartheta_{i1} = m_{i1}l_{ic1}^2 + m_{i2}(l_{i1}^2 + l_{ic2}^2) + J_{i1} + J_{i2}$, $\vartheta_{i2} = m_{i2}l_{i1}l_{ic2}$, $\vartheta_{i3} = m_{i2}l_{ic2}^2 + J_{i2}$, $\vartheta_{i4} = m_{i1}l_{ic1} + m_{i2}l_{i1}$ and $\vartheta_{i5} = m_{i2}l_{ic2}$. In the simulations, let $m_{i1} = 0.4$ kg, $m_{i2} = 0.8$ kg, $l_{i1} = 0.8$ m, $l_{i2} = 1.2$ m, $J_{i1} = 0.0213$ kg m^2, $J_{i2} = 0.096$ kg m^2, $l_{ic1} = \frac{1}{2}l_{i1}$ and $l_{ic2} = \frac{1}{2}l_{i2}$ for any $i \in \mathcal{V}$. Assume that for each $i \in \mathcal{V}$, $m_{i1}, m_{i2}, l_{i1}, l_{i2}, l_{ic1}, l_{ic2}, J_{i1}$ and J_{i2} are unknown. Let $\vartheta_i = \big[m_{i1}l_{ic1}^2 + m_{i2}(l_{i1}^2 + l_{ic2}^2) + J_{i1} + J_{i2}, m_{i2}l_{i1}l_{ic2}, m_{i2}l_{ic2}^2 + J_{i2}, m_{i1}l_{ic1} + m_{i2}l_{i1}, m_{i2}l_{ic2}\big]^T \in \mathbb{R}^5$. From Property 3, it holds that for any $x = \big[x_1, x_2\big]^T$ and $y = \big[y_1, y_2\big]^T$, $Y_i(q_i, \dot{q}_i, y, x) \in \mathbb{R}^{2 \times 5} = \big[Y_{i1}(q_i, \dot{q}_i, y, x) \, Y_{i1}(q_i, \dot{q}_i, y, x)\big]^T$ where $Y_{i1}(q_i, \dot{q}_i, y, x) = \big[x_1, \cos(q_{i2})(2x_1 + x_2) - \sin(q_{i2})(\dot{q}_{i2}y_1 + \dot{q}_{i1}y_2 + \dot{q}_{i2}y_2), x_2, g\cos(q_{i1}), g\cos(q_{i1} + q_{i2})\big]^T$ and $Y_{i2}(q_i, \dot{q}_i, y, x) = \big[0, \cos(q_{i2})x_1 + \sin(q_{i2})\dot{q}_{i1}y_1, x_1 + x_2, 0, g\cos(q_{i1} + q_{i2})\big]^T$.

Each agent $i \in \mathcal{V}$ has a local cost function $f_i(q, t) = [q_{i1} - 0.1i\sin(t)]^2 + [q_{i2} - 0.1i\cos(t)]^2$, and denote by $q^* = \big[q_1^*, q_2^*\big]^T$ the optimal trajectory that minimizes the sum of all the local cost functions $\sum_{i=1}^{10} f_i(q, t)$. In the following algorithm validations, the initial values are chosen as follows: for any $i \in \mathcal{V}$ and any $j \in \{1, 2\}$, $q_{ij}(0)$ and $\dot{q}_{ij}(0)$ are generated randomly from the ranges $[-0.5, 0.5]$, and $v_i(0) = \dot{q}_i(0) + 0.1\mathbf{1}_2$, and $\vartheta_i(0) = \mathbf{0}_5$.
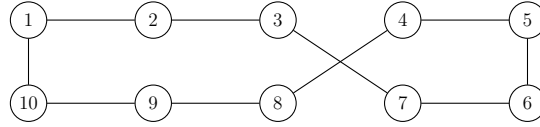
We first validate the distributed time-varying optimization algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.2). The interaction among these ten agents are described by a time-invariant graph given by Graph 1 in Fig. 4.1. In this simulation, we select $\Gamma_i = 0.5I_5$ and $K_i = 25I_2$ for any $i \in \mathcal{V}$, $\alpha = 0.3$, $\beta = 1$ and $\gamma = 8$. The position trajectories and control torques are presented in Fig. 4.2 and Fig. 4.3, respectively. From Fig. 4.2, it shows that all the agents track the optimal trajectory, i.e., $q_i(t) - q^*(t) \to \mathbf{0}_2 \; \forall i \in \mathcal{V}$. It can be seen from Fig. 4.3 that there exists chattering.
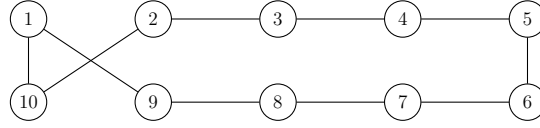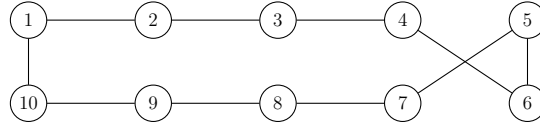
(a) Graph 1.
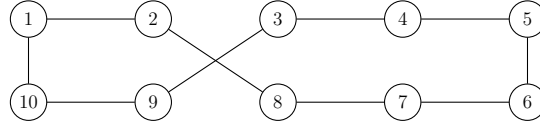
(b) Graph 2.

(c) Graph 3.

(d) Graph 4.

Figure 4.1: Illustration of a switching interaction graph among the ten agents.

We then validate the distributed time-varying optimization algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.18). We select $\Gamma_i = 0.4I_5$ and $K_i = 12I_2$ for any $i \in \mathcal{V}$, $\alpha = 0.5$, $\beta = 1.5$, $\gamma = 7$ and $\varepsilon = 0.5$. The position trajectories and control torques are presented in Fig. 4.4 and Fig. 4.5, respectively. From Fig. 4.4, it shows that all the agents track the optimal trajectory with bounded errors, i.e., $q_i - q^* \in \mathcal{L}_\infty^2$ $\forall i \in \mathcal{V}$. It can seen from Fig. 4.5 that the control torques are smooth and the chattering is removed.
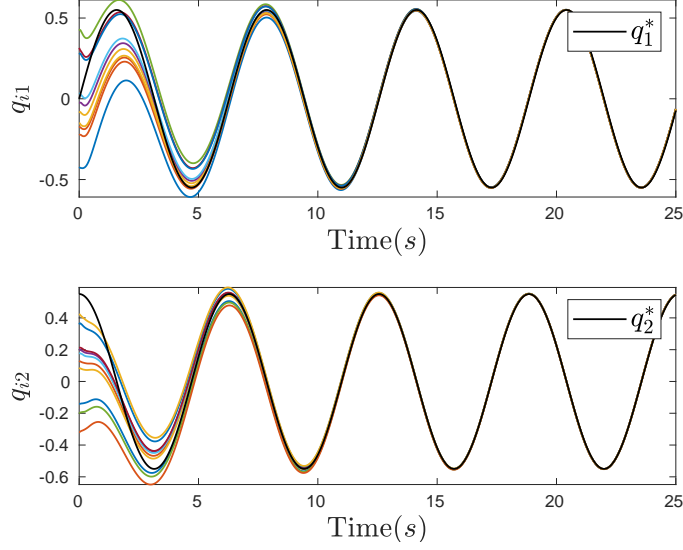
93

Figure 4.2: The position trajectories of Lagrangian agents (1.1) by using the distributed time-varying optimization algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.2). The black lines are the optimal trajectories for each dimension, and the rest are the trajectories of q_{i_1} and q_{i_2}, $i = 1, \ldots, 10$.

We use the same set of cost function to validate the algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.24). The interaction among these ten agents are described by a switching graph shown in Fig. 4.1. The interaction graph states from Graph 1. Then after every 0.25 seconds, it switches to the next graph and the process repeats. We use the same setting for the initial values, and select $\Gamma_i = 0.09I_5$ and $K_i = 14I_2$ for all $i \in \mathcal{V}$, $\alpha = 1$ and $\gamma = 25$. The position trajectories are presented in Fig. 4.6. It can be seen from Fig. 4.6 that the agents track the optimal trajectory, i.e., $q_i(t) - q^*(t) \to \mathbf{0}_2 \; \forall i \in \mathcal{V}$.
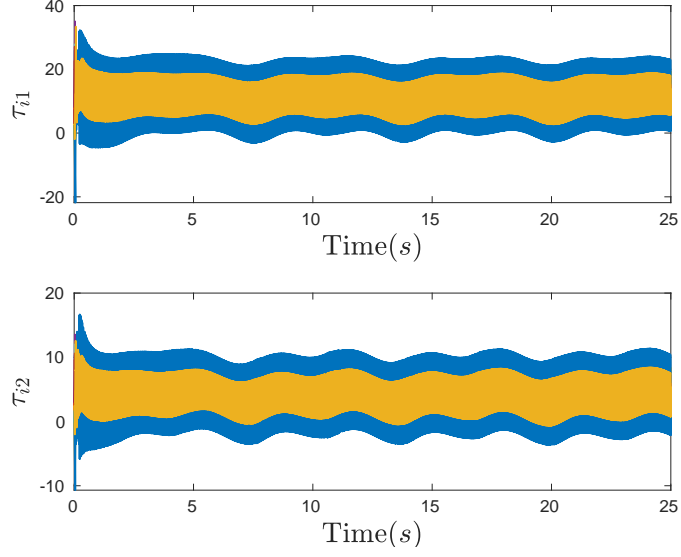
Figure 4.3: The control torques of Lagrangian agents (1.1) by using the distributed time-varying optimization algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.2).
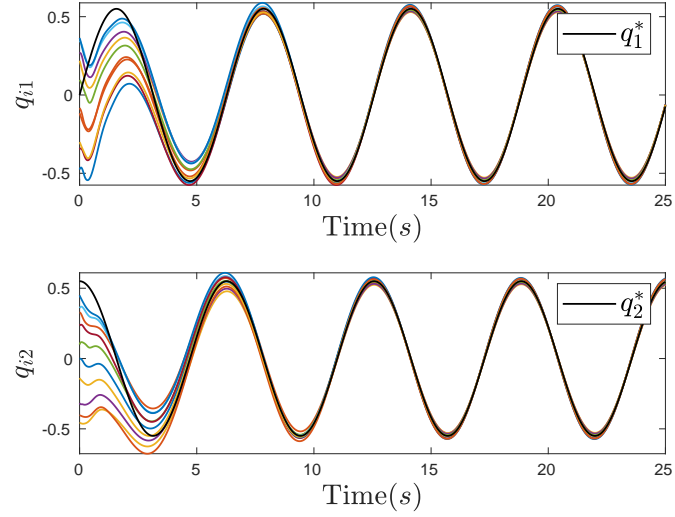

Figure 4.4: The position trajectories of Lagrangian agents (1.1) by using the distributed time-varying optimization algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.18). The black lines are the optimal trajectories for each dimension, and the rest are the trajectories of q_{i_1} and q_{i_2}, $i = 1, \ldots, 10$.
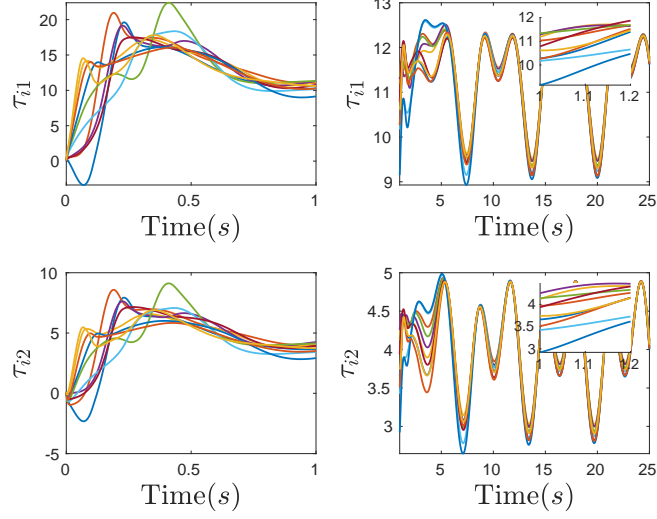
Figure 4.5: The control torques of Lagrangian agents (1.1) by using the distributed time-varying optimization algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.18).
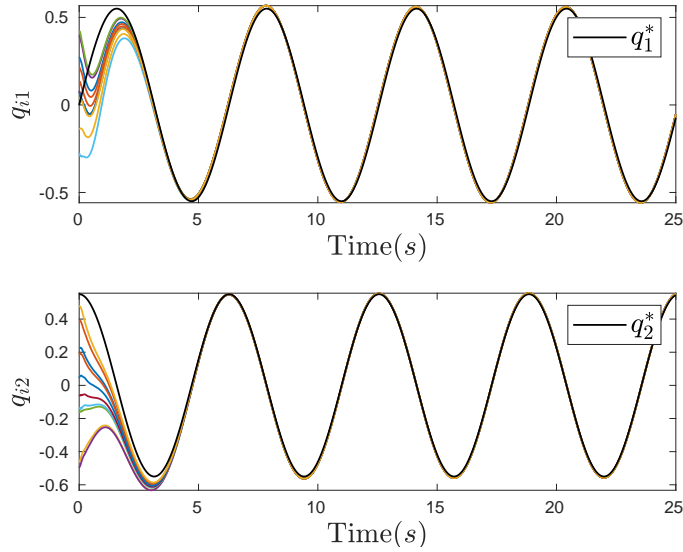

Figure 4.6: The position trajectories of Lagrangian agents (1.1) by using the distributed time-varying optimization algorithm (4.6)-(4.7) with $\dot{v}_i$ defined in (4.24). The black lines are the optimal trajectories for each dimension, and the rest are the trajectories of q_{i_1} and q_{i_2}, $i = 1, \ldots, 10$.

Chapter 5

Conclusions

This dissertation has investigated the following problems:

1. Sampled-data containment control for double-integrator agents with dynamic leaders with nonzero inputs,

2. Robust distributed average tracking for double-integrator agents without velocity measurements under event-triggered communication,

3. Distributed time-varying optimization of networked Lagrangian systems.

Firstly, we proposed a sampled-data based containment control algorithm for a group of double-integrator agents under directed communication networks. This algorithm contributes the solution to the discrete-time containment control problem with dynamic leaders whose inputs are nonzero. It has been shown that, by applying the proposed containment control algorithm, the containment control problem is solved with bounded position and velocity containment control errors, and the ultimate bound of the overall containment control error is proportional to the sampling period.

Secondly, we investigated the distributed average tracking problem for double-integrator agents without velocity measurements under event-triggered communication. First, a base algorithm has been proposed, which remove the dependence of the design parameters' lower bounds on global information. Built on the base algorithm, an event-triggered distributed average tracking algorithm has been designed to remove the continuous communication requirement. The event-triggered algorithm is developed with a new adaptation law and a new triggering condition which overcomes several practical limitations. In addition, a continuous nonlinear function is used approximate the signum function to reduce the chattering phenomenon in reality.

Finally, we have investigated the distributed time-varying optimization of networked Lagrangian systems with parametric uncertainties. The proposed optimization algorithms can be implemented by using only on-board sensors and drive the agents to track the optimal trajectory. First, a base optimization algorithm has been designed to achieve zero optimum-tracking error under fixed graphs. Built on the base optimization algorithm, a continuous variant has been developed, which is capable of generating continuous control torques for the networked Lagrangian systems and hence reducing the chattering. Then, by using the structure of the base optimization algorithm, a distributed time-varying optimization algorithm has been designed under switching graphs.

Bibliography

[1] Rosario Aragues, Jorge Cortes, and Carlos Sagues. Distributed consensus on robot networks for dynamically merging feature-based maps. *IEEE Transactions on Robotics*, 28(4):840–854, 2012.

[2] Mostafa Asgari and Hajar Atrianfar. Necessary and sufficient conditions for containment control of heterogeneous linear multi-agent systems with fixed time delay. *IET Control Theory & Applications*, 13(13):2065–2074, 2019.

[3] He Bai, Randy A Freeman, and Kevin M Lynch. Robust dynamic average consensus of time-varying inputs. In *Proceedings of the IEEE Conference on Decision and Control*, pages 3104–3109. IEEE, 2010.

[4] He Bai, Randy A Freeman, and Kevin M Lynch. Distributed kalman filtering using the internal model average consensus estimator. In *Proceedings of the 2011 American Control Conference*, pages 1500–1505. IEEE, 2011.

[5] Andrey Bernstein, Emiliano Dall'Anese, and Andrea Simonetto. Online primal-dual methods with measurement feedback for time-varying convex optimization. *IEEE Transactions on Signal Processing*, 67(8):1978–1991, 2019.

[6] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[7] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. Siam, 1994.

[8] Yongcan Cao and Wei Ren. Containment control with multiple stationary or dynamic leaders under a directed interaction graph. In *Proceedings of the IEEE Conference on Decision and Control, and the Chinese Control Conference*, pages 3014–3019, 2009.

[9] Yongcan Cao, Wei Ren, and Magnus Egerstedt. Distributed containment control with multiple stationary or dynamic leaders in fixed and switching directed networks. *Automatica*, 48(8):1586–1597, 2012.

[10] Yongcan Cao, Wei Ren, and Yan Li. Distributed discrete-time coordinated tracking with a time-varying reference state and limited communication. *Automatica*, 45(5):1299–1305, 2009.

[11] Yongcan Cao, Daniel Stuart, Wei Ren, and Ziyang Meng. Distributed containment control for multiple autonomous vehicles with double-integrator dynamics: algorithms and experiments. *IEEE Transactions on Control Systems Technology*, 19(4):929–938, 2011.

[12] Yongcan Cao, Wenwu Yu, Wei Ren, and Guanrong Chen. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial informatics*, 9(1):427–438, 2013.

[13] Chien Chern Cheah, Saing Paul Hou, and Jean Jacques E Slotine. Region-based shape control for a swarm of robots. *Automatica*, 45(10):2406–2411, 2009.

[14] Fei Chen, Yongcan Cao, and Wei Ren. Distributed average tracking of multiple time-varying reference signals with bounded derivatives. *IEEE Transactions on Automatic Control*, 57(12):3169–3174, 2012.

[15] Fei Chen, Gang Feng, Lu Liu, and Wei Ren. Distributed average tracking of networked euler-lagrange systems. *IEEE Transactions on Automatic Control*, 60(2):547–552, 2014.

[16] Fei Chen, Gang Feng, Lu Liu, and Wei Ren. Distributed average tracking of networked Euler-Lagrange systems. *IEEE Transactions on Automatic Control*, 60(2):547–552, 2015.

[17] Fei Chen and Wei Ren. Robust distributed average tracking for coupled general linear systems. pages 6953–6958. IEEE, 2013.

[18] Fei Chen and Wei Ren. A connection between dynamic region-following formation control and distributed average tracking. *IEEE transactions on cybernetics*, 48(6):1760–1772, 2017.

[19] Fei Chen and Wei Ren. A connection between dynamic region-following formation control and distributed average tracking. *IEEE Transactions on Cybernetics*, 48(6):1760–1772, 2018.

[20] Fei Chen, Wei Ren, Weiyao Lan, and Guanrong Chen. Tracking the average of time-varying nonsmooth signals for double-integrator agents with a fixed topology. In *Proceedings of the American Control Conference*, pages 2032–2037. IEEE, 2013.

[21] Fei Chen, Wei Ren, Weiyao Lan, and Guanrong Chen. Distributed average tracking for reference signals with bounded accelerations. *IEEE Transactions on Automatic Control*, 60(3):863–869, 2015.

[22] Wei Chen, Derui Ding, Xiaohua Ge, Qing-Long Han, and Guoliang Wei. $\mathcal{H}_\infty$ containment control of multiagent systems under event-triggered communication scheduling: The finite-horizon case. *IEEE Transactions on Cybernetics*, 2018.

[23] Long Cheng, Yunpeng Wang, Wei Ren, Zeng-Guang Hou, and Min Tan. Containment control of multiagent systems with dynamic leaders based on a PI^n-type approach. *IEEE Transactions on Cybernetics*, 46(12):3004–3017, 2016.

[24] Teng-Hu Cheng, Zhen Kan, Justin R Klotz, John M Shea, and Warren E Dixon. Decentralized event-triggered control of networked systems-part 2: Containment control. In *Proceedings of the American Control Conference*, pages 5444–5448, 2015.

[25] Ashish Cherukuri and Jorge Cortes. Initialization-free distributed coordination for economic dispatch under varying loads and generator commitment. *Automatica*, 74:183–193, 2016.

[26] Jorge Cortes. Discontinuous dynamical systems. *IEEE Control systems magazine*, 28(3):36–73, 2008.

[27] Jorge Cortés. Distributed algorithms for reaching consensus on general functions. *Automatica*, 44(3):726–737, 2008.

[28] Jorge Cortés, Geir E Dullerud, Shuo Han, Jerome Le Ny, Sayan Mitra, and George J Pappas. Differential privacy in control and network systems. In *Proceedings of the IEEE Conference on Decision and Control*, pages 4252–4272. IEEE, 2016.

[29] Zhenhua Deng and Yiguang Hong. Multi-agent optimization design for autonomous lagrangian systems. *Unmanned Systems*, 4(01):5–13, 2016.

[30] Yong Ding, Wei Ren, and Yu Zhao. Robust event-triggered distributed average tracking for double-integrator agents without velocity measurements. submitted to the 59th IEEE Conference on Decision and Control.

[31] Yong Ding, Hanlei Wang, and Wei Ren. Distributed continuous-time optimization for networked lagrangian systems with time-varying cost functions under fixed graphs. submitted to the 2022 American Control Conference.

[32] Cynthia Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19, 2008.

[33] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284, 2006.

[34] Christopher Edwards and Sarah Spurgeon. *Sliding mode control: theory and applications*. Crc Press, 1998.

[35] Xiao Feng and Wang Long. Reaching agreement in finite time via continuous local state feedback. pages 711–715. IEEE, 2007.

[36] Aleksei Fedorovich Filippov. *Differential equations with discontinuous righthand sides*. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1988.

[37] Aleksei Fedorovich Filippov. *Differential equations with discontinuous righthand sides: control systems*, volume 18. Springer Science & Business Media, 2013.

[38] Harley Flanders. Differentiation under the integral sign. *The American Mathematical Monthly*, 80(6):615–627, 1973.

[39] Evelyn Frank. On the zeros of polynomials with complex coefficients. *Bulletin of the American Mathematical Society*, 52(2):144–157, 1946.

[40] Randy A Freeman, Peng Yang, and Kevin M Lynch. Stability and convergence properties of dynamic average consensus estimators. In *Proceedings of the IEEE Conference on Decision and Control*, pages 338–343. IEEE, 2006.

[41] Luca Galbusera, Giancarlo Ferrari-Trecate, and Riccardo Scattolini. A hybrid model predictive control scheme for containment and distributed sensing in multi-agent systems. *Systems & Control Letters*, 62(5):413–419, 2013.

[42] Huan Gao and Yongqiang Wang. Dynamics based privacy protection for average consensus on directed graphs. *arXiv preprint arXiv:1812.02255*, 2018.

[43] Jemin George and Randy Freeman. Robust dynamic average consensus algorithms. *IEEE Transactions on Automatic Control*, 64(11):4615–4622, 2019.

[44] Jemin George, Randy A Freeman, and Kevin M Lynch. Robust dynamic average consensus algorithm for signals with bounded derivatives. In *Proceedings of the American Control Conference*, pages 352–357. IEEE, 2017.

[45] Jemin George, Xinlei Yi, and Tao Yang. Distributed robust dynamic average consensus with dynamic event-triggered communication. In *Proceedings of the IEEE Conference on Decision and Control*, pages 434–439. IEEE, 2018.

[46] Sheida Ghapani, Jie Mei, Wei Ren, and Yongduan Song. Fully distributed flocking with a moving leader for lagrange networks with parametric uncertainties. *Automatica*, 67:67–76, 2016.

[47] Sheida Ghapani, Salar Rahili, and Wei Ren. Distributed average tracking for second-order agents with nonlinear dynamics. In *Proceedings of the American Control Conference*, pages 4636–4641. IEEE, 2016.

[48] Sheida Ghapani, Salar Rahili, and Wei Ren. Distributed average tracking of physical second-order agents with heterogeneous unknown nonlinear dynamics without constraint on input signals. *IEEE Transactions on Automatic Control*, 64(3):1178–1184, 2018.

[49] Sheida Ghapani, Wei Ren, and Fei Chen. Distributed average tracking for double-integrator agents without using velocity measurements. In *Proceedings of the American Control Conference*, pages 1445–1450, 2015.

[50] Sheida Ghapani, Wei Ren, Fei Chen, and Yongduan Song. Distributed average tracking for double-integrator multi-agent systems with reduced requirement on velocity measurements. *Automatica*, 81:1–7, 2017.

[51] Silvia Giannini, Antonio Petitti, Donato Di Paola, and Alessandro Rizzo. Asynchronous max-consensus protocol with time delays: Convergence results and applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(2):256–264, 2016.

[52] Antoine Girard. Dynamic triggering mechanisms for event-triggered control. *IEEE Transactions on Automatic Control*, 60(7):1992–1997, 2015.

[53] Musad A Haque, Magnus Egerstedt, and Clyde F Martin. First-order, networked control models of swarming silkworm moths. In *Proceedings of the American Control Conference*, pages 3798–3803, 2008.

[54] Huifen Hong, Wenwu Yu, Xinghuo Yu, Guanghui Wen, and Ahmed Alsaedi. Fixed-time connectivity-preserving distributed average tracking for multiagent systems. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 64(10):1192–1196, 2017.

[55] Yiguang Hong, Jiangping Hu, and Linxin Gao. Tracking control for multi-agent consensus with an active leader and variable topology. *Automatica*, 42(7):1177–1182, 2006.

[56] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[57] Bomin Huang, Yao Zou, Ziyang Meng, and Wei Ren. Distributed time-varying convex optimization for a class of nonlinear multiagent systems. *IEEE Transactions on Automatic Control*, 65(2):801–808, 2020.

[58] Zhenqi Huang, Sayan Mitra, and Geir Dullerud. Differentially private iterative synchronous consensus. In *Proceedings of the 2012 ACM workshop on Privacy in the electronic society*, pages 81–90. ACM, 2012.

[59] Zhenqi Huang, Yu Wang, Sayan Mitra, and Geir E Dullerud. On the cost of differential privacy in distributed control systems. In *Proceedings of the 3rd international conference on High confidence networked systems*, pages 105–114. ACM, 2014.

[60] Meng Ji, Giancarlo Ferrari-Trecate, Magnus Egerstedt, and Annalisa Buffa. Containment control in mobile networks. *IEEE Transactions on Automatic Control*, 53(8):1972–1975, 2008.

[61] Zhong-Ping Jiang and Yuan Wang. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37(6):857–869, 2001.

[62] Hassan K Khalil. *Nonlinear systems*. New Jersey: Prentice Hall, 2002.

[63] Solmaz S Kia, Jorge Cortés, and Sonia Martínez. Singularly perturbed algorithms for dynamic average consensus. pages 1758–1763. IEEE, 2013.

[64] Solmaz S Kia, Jorge Cortés, and Sonia Martínez. Distributed event-triggered communication for dynamic average consensus in networked systems. *Automatica*, 59:112–119, 2015.

[65] Solmaz S Kia, Jorge Cortés, and Sonia Martinez. Dynamic average consensus under limited control authority and privacy requirements. *International Journal of Robust and Nonlinear Control*, 25(13):1941–1966, 2015.

[66] Istvan Kovacs, Daniel S Silver, and Susan G Williams. Determinants of block matrices and schurs formula. *American Mathematical Monthly*, 106(5):950–952, 1999.

[67] Joseph P LaSalle. *The stability and control of discrete processes*, volume 62. Springer Science & Business Media, 2012.

[68] Jianzhen Li, Wei Ren, and Shengyuan Xu. Distributed containment control with multiple dynamic leaders for double-integrator dynamics using only position measurements. *IEEE Transactions on Automatic Control*, 57(6):1553–1559, 2011.

[69] Zhongkui Li, Xiangdong Liu, Wei Ren, and Lihua Xie. Distributed tracking control for linear multiagent systems with a leader of bounded unknown input. *IEEE Transactions on Automatic Control*, 58(2):518–523, 2013.

[70] Zhongkui Li, Wei Ren, Xiangdong Liu, and Mengyin Fu. Distributed containment control of multi-agent systems with general linear dynamics in the presence of multiple leaders. *International Journal of Robust and Nonlinear Control*, 23(5):534–547, 2013.

[71] Qing Ling and Alejandro Ribeiro. Decentralized dynamic optimization through the alternating direction method of multipliers. *IEEE Transactions on Signal Processing*, 62(5):1185–1197, 2014.

[72] Huiyang Liu, Guangming Xie, and Long Wang. Necessary and sufficient conditions for containment control of networked multi-agent systems. *Automatica*, 48(7):1415–1422, 2012.

[73] Huiyang Liu, Guangming Xie, and Mei Yu. Necessary and sufficient conditions for containment control of fractional-order multi-agent systems. *Neurocomputing*, 323:86–95, 2019.

[74] Kaien Liu, Zhijian Ji, Guangming Xie, and Ruiping Xu. Event-based broadcasting containment control for multi-agent systems under directed topology. *International Journal of Control*, 89(11):2360–2370, 2016.

[75] Wenhui Liu, Chunjie Yang, Youxian Sun, and Jiaxiang Qin. Observer-based event-triggered containment control of multi-agent systems with time delay. *International Journal of Systems Science*, 48(6):1217–1225, 2017.

[76] Hui Lü, Wangli He, Qing-Long Han, Xiaohua Ge, and Chen Peng. Finite-time containment control for nonlinear multi-agent systems with external disturbances. *Information Sciences*, 2019.

[77] JL Mancilla-Aquilar and Rafael Antonio Garcia. On the existence of common lyapunov triples for iss and iiss switched systems. In *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, volume 4, pages 3507–3512. IEEE, 2000.

[78] Jie Mei, Wei Ren, and Guangfu Ma. Distributed containment control for lagrangian networks with parametric uncertainties under a directed graph. *Automatica*, 48(4):653–659, 2012.

[79] Ziyang Meng, Wei Ren, and Zheng You. Distributed finite-time attitude containment control for multiple rigid bodies. *Automatica*, 46(12):2092–2099, 2010.

[80] Guoying Miao, Jinde Cao, Ahmed Alsaedi, and Fuad E Alsaadi. Event-triggered containment control for multi-agent systems with constant time delays. *Journal of the Franklin Institute*, 354(15):6956–6977, 2017.

[81] Yilin Mo and Richard M Murray. Privacy preserving average consensus. *IEEE Transactions on Automatic Control*, 62(2):753–765, 2017.

[82] Hossein Moradian and Solmaz S Kia. On robustness analysis of a dynamic average consensus algorithm to communication delay. *IEEE Transactions on Control of Network Systems*, 6(2):633–641, 2018.

[83] KS Narendra and JD Boskovic. A combined direct, indirect, and variable structure method for robust adaptive control. *IEEE Transactions on Automatic Control*, 37(2):262–268, 1992.

[84] S Nosrati, M Shafiee, and MB Menhaj. Synthesis and analysis of robust dynamic linear protocols for dynamic average consensus estimators. *IET control theory & applications*, 3(11):1499–1516, 2009.

[85] Shahram Nosrati, Masoud Shafiee, and Mohammad Bagher Menhaj. Dynamic average consensus via nonlinear protocols. *Automatica*, 48(9):2262–2270, 2012.

[86] Erfan Nozari, Pavankumar Tallapragada, and Jorge Cortés. Differentially private average consensus: Obstructions, trade-offs, and optimal algorithm design. *Automatica*, 81:221–231, 2017.

[87] Erfan Nozari, Pavankumar Tallapragada, and Jorge Cortés. Differentially private distributed convex optimization via functional perturbation. *IEEE Transactions on Control of Network Systems*, 5(1):395–408, 2018.

[88] Petter Ogren, Edward Fiorelli, and Naomi Ehrich Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, 2004.

[89] Reza Olfati-Saber and Richard M Murray. Distributed cooperative control of multiple vehicle formations using structural potential functions. *IFAC Proceedings Volumes*, 35(1):495–500, 2002.

[90] Reza Olfati-Saber and Richard M Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[91] Reza Olfati-Saber and Jeff S Shamma. Consensus filters for sensor networks and distributed sensor fusion. In *Proceedings of the IEEE Conference on Decision and Control*, pages 6698–6703, 2005.

[92] Salar Rahili and Wei Ren. Distributed continuous-time convex optimization with time-varying cost functions. *IEEE Transactions on Automatic Control*, 62(4):1590–1605, 2017.

[93] Salar Rahili and Wei Ren. Distributed continuous-time convex optimization with time-varying cost functions. *IEEE Transactions on Automatic Control*, 62(4):1590–1605, 2017.

[94] Wei Ren and Randal W Beard. Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control*, 50(5):655–661, 2005.

[95] Wei Ren, Randal W Beard, and Ella M Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems Magazine*, 27(2):71–82, 2007.

[96] Minghao Ruan, Huan Gao, and Yongqiang Wang. Secure and privacy-preserving consensus. *IEEE Transactions on Automatic Control*, 2019.

[97] Muhammad Saim, Sheida Ghapani, Wei Ren, Khalid Munawar, and Ubaid M Al-Saggaf. Distributed average tracking in multi-agent coordination: Extensions and experiments. *IEEE Systems Journal*, 12(3):2428–2436, 2017.

[98] Jinliang Shao, Lei Shi, Wei Xing Zheng, and Yuhua Cheng. Cooperative containment control in time-delayed multi-agent systems with discrete-time high-order dynamics under dynamically changing topologies. *Journal of the Franklin Institute*, 356(5):2441–2462, 2019.

[99] Jinliang Shao, Lei Shi, Wei Xing Zheng, and Ting-Zhu Huang. Containment control for heterogeneous multi-agent systems with asynchronous updates. *Information Sciences*, 436:74–88, 2018.

[100] Daniel Shevitz and Brad Paden. Lyapunov stability theory of nonsmooth systems. *IEEE Transactions on automatic control*, 39(9):1910–1914, 1994.

[101] Andrea Simonetto, Leon Kester, and Geert Leus. Distributed time-varying stochastic optimization and utility-based communication. *arXiv preprint arXiv:1408.5294*, 2014.

[102] Andrea Simonetto, Aryan Mokhtari, Alec Koppel, Geert Leus, and Alejandro Ribeiro. A class of prediction-correction methods for time-varying convex optimization. *IEEE Transactions on Signal Processing*, 64(17):4576–4591, 2016.

[103] Jean-Jacques E Slotine and Weiping Li. *Applied nonlinear control*, volume 199. Prentice hall Englewood Cliffs, NJ, 1991.

[104] Demetri P Spanos, Reza Olfati-Saber, and Richard M Murray. Dynamic consensus on mobile networks. In *IFAC World Congress*, pages 1–6. Citeseer, 2005.

[105] Mark W Spong, Seth Hutchinson, and Mathukumalli Vidyasagar. *Robot modeling and control*. New Jersey: Wiley, 2006.

[106] Chao Sun, Maojiao Ye, and Guoqiang Hu. Distributed time-varying quadratic optimization for multiple agents under undirected graphs. *IEEE Transactions on Automatic Control*, 62(7):3687–3694, 2017.

[107] Shan Sun and Wei Ren. Distributed continuous-time optimization with time-varying objective functions and inequality constraints. In *Proceedings of the IEEE Conference on Decision and Control*, pages 5622–5627. IEEE, 2020.

[108] Petr Švestka and Mark H Overmars. Coordinated path planning for multiple robots. *Robotics and autonomous systems*, 23(3):125–152, 1998.

[109] Damiano Varagnolo, Gianluigi Pillonetto, and Luca Schenato. Distributed cardinality estimation in anonymous networks. *IEEE Transactions on Automatic Control*, 59(3):645–659, 2013.

[110] Richard Vaughan, Neil Sumpter, Jane Henderson, Andy Frost, and Stephen Cameron. Experiments in automatic flock control. *Robotics and Autonomous Systems*, 31(1-2):109–117, 2000.

[111] Linh Vu, Debasish Chatterjee, and Daniel Liberzon. Input-to-state stability of switched systems and switching adaptive control. *Automatica*, 43(4):639–646, 2007.

[112] Dong Wang and Wei Wang. Necessary and sufficient conditions for containment control of multi-agent systems with time delay. *Automatica*, 103:418–423, 2019.

[113] Hanlei Wang. Differential-cascade framework for consensus of networked lagrangian systems. *Automatica*, 112:108620, 2020.

[114] Hanlei Wang, Wei Ren, and Chien Chern Cheah. Forwardstepping: A new approach for control of dynamical systems. In *Proceedings of the American Control Conference*, pages 1208–1215. IEEE, 2019.

[115] Hanlei Wang, Wei Ren, and Chien Chern Cheah. A differential-cascaded paradigm for control of nonlinear systems. *arXiv preprint arXiv:2012.14251*, 2020.

[116] Xiaoping Wang, Lei Shi, and Jingliang Shao. Containment control for high-order multi-agent systems with heterogeneous time delays. *IET Control Theory & Applications*, 12(9):1340–1348, 2018.

[117] Yu Wang, Zhenqi Huang, Sayan Mitra, and Geir E Dullerud. Differential privacy in linear distributed control systems: Entropy minimizing mechanisms and performance tradeoffs. *IEEE Transactions on Control of Network Systems*, 4(1):118–130, 2017.

[118] Hong Xia, Wei Xing Zheng, and Jinliang Shao. Event-triggered containment control for second-order multi-agent systems with sampled position data. *ISA Transactions*, 73:91–99, 2018.

[119] Guangming Xie, Huiyang Liu, Long Wang, and Yingmin Jia. Consensus in networked multi-agent systems via sampled control: fixed topology case. In *Proceedings of the American Control Conference*, pages 3902–3907. IEEE, 2009.

[120] Quan Xiong, Peng Lin, Zhiyong Chen, Wei Ren, and Weihua Gui. Distributed containment control for first-order and second-order multiagent systems with arbitrarily bounded delays. *International Journal of Robust and Nonlinear Control*, 29(4):1122–1131, 2019.

[121] Quan Xiong, Peng Lin, Wei Ren, Chunhua Yang, and Weihua Gui. Containment control for discrete-time multiagent systems with communication delays and switching topologies. *IEEE Transactions on Cybernetics*, (99):1–4, 2018.

[122] Peng Yang, Randy A Freeman, and Kevin M Lynch. Multi-agent coordination by decentralized estimation and control. *IEEE Transactions on Automatic Control*, 53(11):2480–2496, 2008.

[123] Tao Yang, Xinlei Yi, Junfeng Wu, Ye Yuan, Di Wu, Ziyang Meng, Yiguang Hong, Hong Wang, Zongli Lin, and Karl H Johansson. A survey of distributed optimization. *Annual Reviews in Control*, 47:278–305, 2019.

[124] Xinlei Yi, Xiuxian Li, Lihua Xie, and Karl H Johansson. Distributed online convex optimization with time-varying coupled inequality constraints. *IEEE Transactions on Signal Processing*, 68:731–746, 2020.

[125] Daniel Zelazo, Amirreza Rahmani, and Mehran Mesbahi. Agreement via the edge laplacian. In *Proceedings of the IEEE Conference on Decision and Control*, pages 2309–2314. IEEE, 2007.

[126] Wenbing Zhang, Yang Tang, Yurong Liu, and Jürgen Kurths. Event-triggering containment control for a class of multi-agent networks with fixed and switching topologies. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 64(3):619–629, 2017.

[127] Yanqiong Zhang, Zhenhua Deng, and Yiguang Hong. Distributed optimal coordination for multiple heterogeneous euler–lagrangian systems. *Automatica*, 79:207–213, 2017.

[128] Yu Zhao, Yongfang Liu, Zhongkui Li, and Zhisheng Duan. Distributed average tracking for multiple signals generated by linear dynamical systems: An edge-based framework. *Automatica*, 75:158–166, 2017.

[129] Yu Zhao, Yongfang Liu, Guanghui Wen, and Tingwen Huang. Finite-time distributed average tracking for second-order nonlinear systems. *IEEE Transactions on Neural Networks and Learning Systems*, 30(6):1780–1789, 2019.

[130] Yu Zhao, Yongfang Liu, Guanghui Wen, Xinghuo Yu, and Guanrong Chen. Distributed average tracking for lipschitz-type of nonlinear dynamical systems. *IEEE Transactions on Cybernetics*, 49(12):4140–4152, 2019.

[131] Wencheng Zou and Zhengrong Xiang. Event-triggered containment control of second-order nonlinear multi-agent systems. *Journal of the Franklin Institute*, 356(17):10421–10438, 2019.

[132] Yao Zou, Bomin Huang, and Ziyang Meng. Distributed continuous-time algorithm for constrained optimization of networked Euler-Lagrange systems. *IEEE Transactions on Control of Network Systems*, 8(2):1034–1042, 2021.

[133] Yao Zou, Ziyang Meng, and Yiguang Hong. Adaptive distributed optimization algorithms for Euler–Lagrange systems. *Automatica*, 119:109060, 2020.