Distributed Model Predictive Control based Consensus of General Linear Multi-agent Systems with Input Contraints

by

Zhuo Li

B. Eng., Northwestern Polytechnical University, 2017

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Mechanical Engineering

University of Victoria

Distributed Model Predictive Control based Consensus of General Linear Multi-agent Systems with Input Contraints

by

Zhuo Li
B. Eng., Northwestern Polytechnical University, 2017

Supervisory Committee

Dr. Yang Shi, Supervisor
(Department of Mechanical Engineering)

Dr. Daniela Constantinescu, Departmental Member
(Department of Mechnical Engineering)

Supervisory Committee

Dr. Yang Shi, Supervisor

(Department of Mechanical Engineering)

Dr. Daniela Constantinescu, Departmental Member

(Department of Mechnical Engineering)

ABSTRACT

In the study of multi-agent systems (MASs), cooperative control is one of the most fundamental issues. As it covers a broad spectrum of applications in many industrial areas, there is a desire to design cooperative control protocols for different system and network setups. Motivated by this fact, in this thesis we focus on elaborating consensus protocol design, via model predictive control (MPC), under two different scenarios: (1) general constrained linear MASs with bounded additive disturbance; (2) linear MASs with input constraints underlying distributed communication networks.

In Chapter 2, a tube-based robust MPC consensus protocol for constrained linear MASs is proposed. For undisturbed linear MASs without constraints, the results on designing a centralized linear consensus protocol are first developed by a suboptimal linear quadratic approach. In order to evaluate the control performance of the suboptimal consensus protocol, we use an infinite horizon linear quadratic objective function to penalize the disagreement among agents and the size of control inputs. Due to the non-convexity of the performance function, an optimal controller gain is

difficult or even impossible to find, thus a suboptimal consensus protocol is derived. In the presence of disturbance, the original MASs may not maintain certain properties such as stability and cooperative performance. To this end, a tube-based robust MPC framework is introduced. When disturbance is involved, the original constraints in nominal prediction should be tightened so as to achieve robust constraint satisfaction, as the predicted states and the actual states are not necessarily the same. Moreover, the corresponding robust constraint sets can be determined offline, requiring no extra iterative online computation in implementation.

In Chapter 3, a novel distributed MPC-based consensus protocol is proposed for general linear MASs with input constraints. For the linear MAS without constraints, a pre-stabilizing distributed linear consensus protocol is developed by an inverse optimal approach, such that the corresponding closed-loop system is asymptotically set stable with respect to a consensus set. Implementing this pre-stabilizing controller in a distributed digital setting is however not possible, as it requires every local decision maker to continuously access the state of their neighbors simultaneously when updating the control input. To relax these requirements, the assumed neighboring state, instead of the actual state of neighbors, is used. In our distributed MPC scheme, each local controller minimizes a group of control variables to generate control input. Moreover, an additional state constraint is proposed to bound deviation between the actual and the assumed state. In this way, consistency is enforced between intended behaviors of an agent and what its neighbors believe it will behave. We later show that the closed-loop system converges to a neighboring set of the consensus set thanks to the bounded state deviation in prediction.

In Chapter 4, conclusions are made and some research topics for future exploring are presented.

Contents

List of Tables

List of Figures

Acronym

MAS Multi-agent system

RHC Receding horizon control

MPC Model predictive control

LQR Linear quadratic regulator

LTI Linear time-invariant

DARE Discrete-time algebraic Riccati equation

OCP Optimal control problem

KKT conditions Karush–Kuhn–Tucker conditions

UAV Unmanned aerial vehicle

AUV Autonomous underwater vehicle

ACKNOWLEDGMENTS

First and foremost I would like to thank my supervisor Dr. Yang Shi, a professional and respectful scholar. It has been a great honor to be his student during the past two years. I sincerely appreciate his insightful advice and comments, leading me to the gate of academic research. His contributions of time, patience and extensive knowledge in research inspire me to move forward and make my MASc experience stimulating. His enthusiasm for research was contagious and motivational for me especially when I was in my hard times in doing research. He is not only a respectful professor who guides me and provides me with opportunities to strengthen my abilities, but also a warm and wise friend who helps me in many perspectives of personal life.

I would like to thank the thesis committee members, Prof. Daniela Constantinescu and Prof. Kui Wu for their constructive comments and suggestions which helped me improve my thesis.

The members in our research group ACIPL have contributed enormously to my professional career and personal life at UVic. Particularly, I would like to thank Dr. Bingxian Mu for picking me up at the airport and helping me settle down when I came to Victoria two years ago. I am also grateful for the role model Dr. Mu has provided as a decent researcher and a young professor. I would like to show my special appreciation to Changxin Liu as well. He shared me with his valuable experience in doing research and his constructive suggestions on academic writing were really helpful to me. I feel very lucky to know Jicheng Chen and Kunwu Zhang, who encouraged me a lot when I was in low spirits.

I also want to thank Dr. Chao Shen, Dr. Yuanye Chen, Henglai Wei, Qian Zhang, Qi Sun, Tianyu Tan, Xinxin Shang, Yuan Yang, Xiang Sheng, Zhang Zhang, Huaiyuan Sheng, Chen Ma, Chonghan Ma, Tianxiang Lu and many other visiting professors and students for their accompany. I benefited a lot from Patrick Chang during my TA working and learned a lot from Prof. Chunxi Yang's feedback in undergraduate course teaching.

I must appreciate my friendships with Xiaoyu Du and Bo Wang in Shanghai for their care and encouragement. Most importantly, I would like to thank my parents for their love and support.

Chapter 1

Introduction

1.1 Cooperative Control of Multi-agent Systems

In the past few decades, cooperative control in MASs, like formation, tracking and consensus, has received increasing attention. It covers a broad spectrum of applications in autonomous vehicles, distributed sensor networks, cyber-physical systems and power grids. Technically, the term "agent" refers to a general individual system dynamic. It can be characterized as a single mobile robot or an unmanned aerial vehicle (UAV) in a multi-robotic system, or a single satellite in a global navigation system, or a photovoltaic panel in a micro power grid. Equipped with actuators and sensors, agents can share information with others via communication networks to perform complex tasks which are difficult or even impossible for a single agent. A general architecture of MASs is shown in Figure 1.1. A cooperative control law is called a *protocol* in the study of MASs. In cooperative control problems, information shared among agents may involve common objectives, common control protocols, relative state information, or topology of communication networks.

One approach to tackle multi-agent cooperative control problem is to implement a

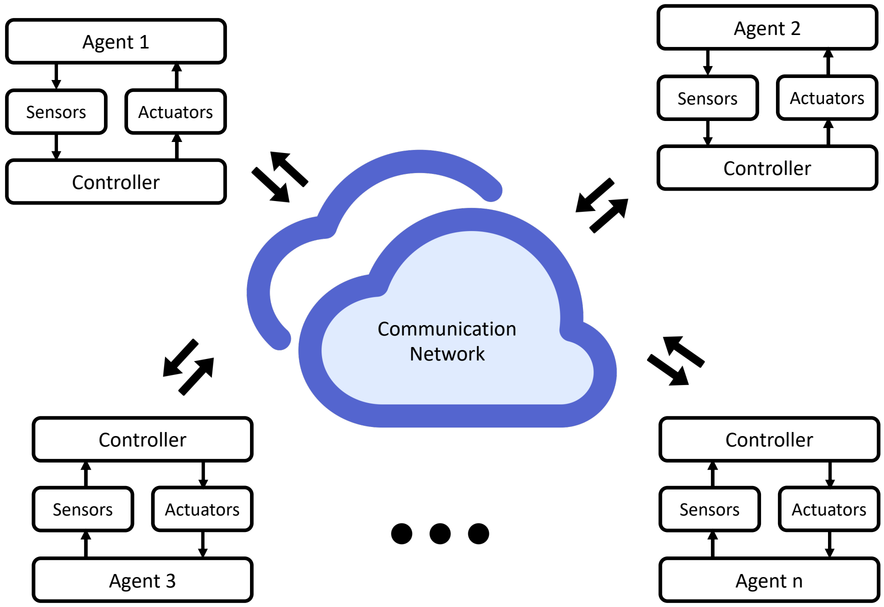


Figure 1.1: A general architecture of MASs

centralized controller to regulate all agents. It aims to use only one computationally powerful central controller to control the overall MAS. After collecting data and calculating, the centralized controller sends control input signals to every agent. With properly designed protocols, the centralized cooperative control strategy works well in the MASs whose scales are not very large.

In many practical applications, the number of agents can grow tremendously, thus computational load and communication pressure on the system may increase. Together with disturbance like sensor noise and model mismatch, the cooperation among agents may fail. Another approach featuring the computational efficiency is to apply decentralized cooperative control frameworks. A large-scale MAS is decoupled into several subsystems by neglecting interactions among agents, and then an independent controller is assigned to every agent to generate control signals.

However, ignorance of interactions among agents is possible to result in poor control performance or even loss of convergence. To this end, increasing attention has

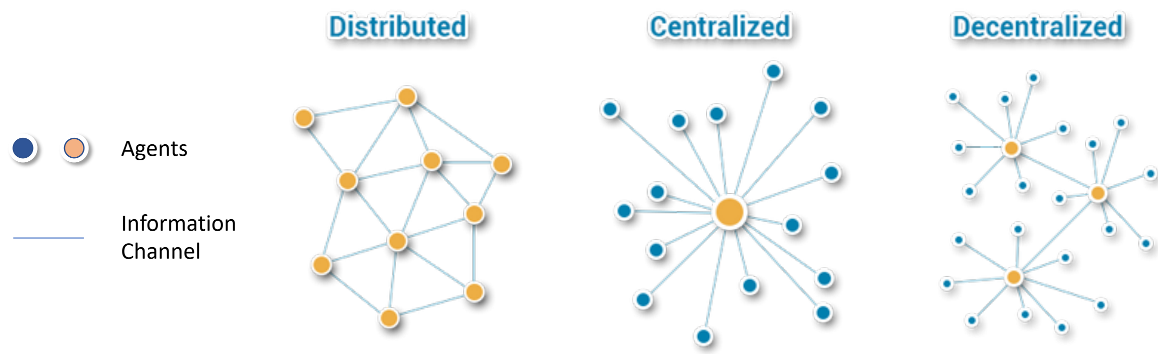


Figure 1.2: Different network architectures

been devoted to effective, but more reliable distributed control schemes. Similar to the structure of the decentralized control strategies, a local controller is assigned to each agent in the distributed control schemes, but the interactions are considered in controller design. Therefore, the distributed control strategies can achieve cooperative control tasks as the centralized ones, as well as reduce the computational complexity due to the decentralized network structure. Figure 1.2 demonstrates the three different communication network architectures.

Applications of multi-agent cooperation are reported in multi-vehicle system formation control [1, 2], leader-follower flocking [3, 4], trajectory tracking [5], point tracking [6] and so forth. The centralized cooperative control strategies can be found in [7, 8], and existing works [9, 10] demonstrate the decentralized cooperative control schemes with robustness to disturbance. The distributed control strategies are also reported in [11, 12, 13]. In the following sections, a brief introduction to multi-agent consensus is given and a literature review is presented to illustrate the recent research progress.

1.2 Consensus Problem in Multi-agent Systems

Multi-agent consensus, also known as multi-agent agreement, requires a group of agents to agree on certain quantities of interest [14]. Consensus problem has a long history in the research field of computer science [15], especially in automata and distributed computation, but in this thesis we focus on its applications from the perspective of automatic control.

In a typical MAS, a group of autonomous agents are equipped with build-in sensors and actuators. Each agent has an embedded controller to generate control inputs individually. The agents measure their states and communicate with other agents via an information transmission network. In this way, the overall system works in a collaborative way. The overall system is said to achieve consensus if all agents reach an agreement on certain common features, such as common equilibrium points, position, linear/angular velocity or orientation [16]. The schematic of a typical multi-agent consensus problem is demonstrated in Figure 1.3.

Appropriate consensus protocols are necessary and crucial to multi-agent cooperation. To elaborate this, we consider the following MAS given by discrete-time linear time-invariant (LTI) dynamics:

$$x_i(k+1) = Ax_i(k) + Bu_i(k), \quad x_i(0) = x_{i0}, \quad i \in \mathbb{N}_{[1,M]},$$

where $x_i \in \mathbb{R}^n$ is the state, $u_i \in \mathbb{R}^m$ is the control input and x_{i0} is the initial state. $\mathbb{N}_{[1,M]}$ represents a sequence of integers $\{1, \ldots, M\}$. The MAS is said to reach consensus if

$$\lim_{k\to\infty} \|x_i(k) - x_j(k)\| = 0,$$

where i, $j \in \mathbb{N}_{[1,M]}$ and $i \neq j$. Without loss of generality, our objective is to design a

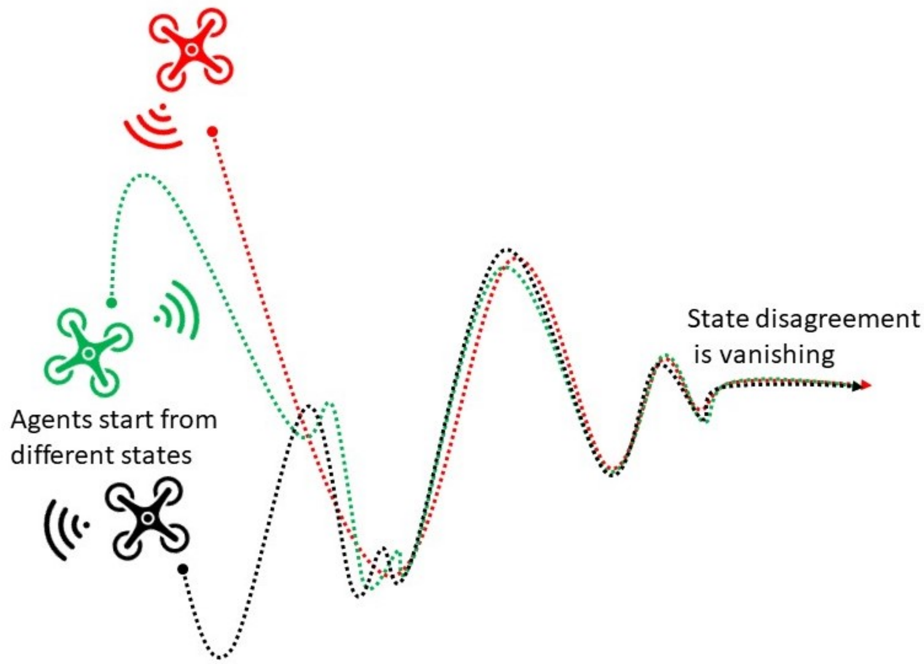


Figure 1.3: Schematic of consensus problem in MASs

consensus protocol of the following form

$$u_i(k) = g(x_i(k), x_j(k)),$$

where $j \in \mathcal{N}_i(k)$ represents the set of neighbor agents whose information is accessible to agent i at time k. Our focus is mainly on fixed network topology, so $\mathcal{N}_i(k)$ is assumed to be time-invariant in this thesis.

The topology of the communication network is described by a *graph*. In this thesis, a graph $\mathcal{G}$ is assumed to be time-invariant and is defined by $(\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{v_1, v_2, \cdots, v_M\}$ being a non-empty vertex set of M nodes and

$$\mathcal{E} = \{(v_i, v_j)\,|v_i, v_j \in \mathcal{V}, i \neq j\} \subset \mathcal{V} \times \mathcal{V}$$

being the edge set. More about graph theory is referred to [16, 17].

Since a consensus control scheme can be easily modified to solve stabilization, formation, leader-follower flocking, trajectory tracking and some other cooperative control problems, it is one of the most fundamental issues in multi-agent cooperation, and has been studied from different perspectives in the past few decades:

- **System dynamics.** In the early stage, simple system dynamics like single-integrator receive major attention, and some decent results can be found in [18, 19, 20]. Extensions for more complex double-integrator systems are made and related works are reported in [21, 22, 23]. Particularly, authors in [24] demonstrate the sufficient and necessary conditions for a second-order MAS reaching consensus under a directed communication graph with a spanning tree. Systems in practical industrial applications are often more complex, thus consensus for more general linear systems is investigated [25, 26, 27, 28]. However, these aforementioned consensus control schemes may not be directly applicable to higher order dynamics or nonlinear systems. Till now, consensus solutions to nonlinear MASs are still few, but some exceptions can be found in [29, 30].

- **Applications.** Multi-agent consensus has found many applications in industry, particularly in UAV control [31, 26], multi-vehicle collaboration [21, 32, 33] or platoon [27] and wireless sensor networks [34]. In the field of power grid management, implementations of MAS consensus are reported in power restoration and distributed power generation [35, 36].

- **Practical constraints.** Challenges are brought to the real applications of multi-agent consensus when practical constraints are involved. Generally, these constraints can be categorized as two types: state constraints and input saturation. Satisfaction of state constraints is often considered in multi-agent cooperation. For example, if the distance between two agents is too closed, collision may happen and cooperation among agents may fail. An example of collision avoidance can be referred to [26]. On the contrary, if the distance is too far, communication connection among agents is possible to lose, so some researchers develop results considering connectivity maintenance [37]. Another category of practical constraints is input saturation. Due to the physical limitations of electrical/mechanical actuators, control input has to be restricted to reasonable levels, e.g. [38, 27].

Some typical results in multi-agent consensus subject to different constraints are categorized in Table 1.1.

Table 1.1: Brief literature review for multi-agent consensus.

Related Work	**System Model**	**Cooperation Type**	**Practical Constraints**
B-Arranz *et al*, 2014	Nonlinear & Continuous	Formation	Distance constraints
Lin *et al*, 2014	Linear & Continuous	Formation	Distance constraints
Zhan *et al*, 2012	Linear & Discrete	Flocking	Distance constraints
Gu *et al*, 2009	Nonlinear & Continuous	Flocking	Input saturation
Li *et al*, 2015	Nonlinear & Continuous	Trajectory tracking	Input saturation
Su *et al*, 2013	Linear & Continuous	Consensus	Input saturation
Kuriki *et al*, 2014	Linear & Discrete	Consensus	Collision avoidance
Li *et al*, 2018	Linear & Discrete	Consensus	Input saturation
Zhan *et al*, 2018	Linear & Discrete	Consensus	Input saturation & Distance constraints

1.3 MPC and MPC-based Consensus

1.3.1 MPC

Model predictive control (MPC), also known as receding horizon control (RHC), is an advanced optimal control strategy combining the feedback mechanism and constrained convex optimization techniques. The control input is generated by solving an optimal control problem (OCP) where the cost functional is a function of the current system state and a sequence of control variables over a certain time horizon in the future prediction. The constraints of the OCPs are designed based on the inherent physical restrictions of real systems. MPC finds its applications in many engineering domains where practical constraints are involved, such as chemical plant control [39, 40], path planning of AUVs [41] and power grids [35, 36].

Consider a discrete-time linear system given by

$$x(k+1) = Ax(k) + Bu(k), \tag{1.1}$$

where $x \in \mathbb{R}^n$ is the state and $u \in \mathbb{R}^m$ is the control input. The system is required to satisfy the state constraints $x \in \mathcal{X}$ and the input constraints $u \in \mathcal{U}$. The cost function of the corresponding OCP to be solved iteratively can be defined as

$$J_N(x(k), \boldsymbol{u}(k)) = \sum_{t=0}^{N-1} \|x(k+t|k)\|_Q^2 + \|u(k+t|k)\|_R^2 + \|x(k+N|k)\|_P^2 \,,$$

where N denotes the prediction horizon, $x(k+t|k)$ and $u(k+t|k)$ represent the predicted state and input trajectories at time $k+t$ starting from time k and satisfy

$$\begin{aligned} &x(k|k) = x(k), \\ &x(k+t+1|k) = Ax(k+t|k) + Bu(k+t|k), \end{aligned} \tag{1.2}$$

and $Q \succeq 0$, $R \succ 0$ and $P \succ 0$ are weighting matrices. The predicted control sequence is defined by $\boldsymbol{u}(k)^{\mathrm{T}} = \left[u(k|k)^{\mathrm{T}}, \cdots, u(k-N|k)^{\mathrm{T}} \right]^{\mathrm{T}}$. Then at time instant k, the control input sequence is obtained by solving the following optimization problem:

$$
\begin{aligned}
\min_{\boldsymbol{u}(\cdot)} \quad & J_N(x(k), \boldsymbol{u}(k)) \\
\text{s.t.} \quad & (1.2) \\
& x(k+t|k) \in \mathcal{X}, \quad u(k+t|k) \in \mathcal{U}, \\
& x(k+N|k) \in \mathcal{X}_f,
\end{aligned}
$$

where the target set $\mathcal{X}_f$ is called the terminal set. In most cases, once the optimal predicted control sequence $\boldsymbol{u}^*(k) = \arg\min J_N(x(k), \boldsymbol{u}(k))$ is obtained, only the first element $u^*(k|k)$ is applied to the actual system. As time moves, the system re-samples the current state and solves the above online optimization problem to generate control signals iteratively.

Compared with other control methods, MPC has proved success in tackling hard constraints in multi-variable control. The process industry witnessed the phenomenal success of MPC at the beginning of this century, but paid less attention to the conditions that guarantee stability of MPC. Fortunately, a breakthrough in deterministic MPC stability study happens in 2000. The researchers in [42] discuss the conditions that ensure nominal stability of linear and nonlinear systems with state and input constraints of the MPC frameworks. It is well understood that the stability of MPC can be achieved by adding a properly designed terminal cost and terminal constraints, or by extending the prediction horizon of the online optimization problem. Many literatures also present robust MPC schemes against additive disturbance. An overview of typical MPC schemes can be summarized as follows:

- To guarantee iterative feasibility and stability in deterministic MPC design, some tailored terminal constraints and terminal state penalty are often added to the online optimization problem in the model predictive controllers. For linear systems, the essential idea of stable MPC frameworks is to find a positive invariant set as the terminal region [42]. When the system state is inside the offline determined terminal set, all constraints are recursively satisfied. A notable work [43] proposes a stabilizing MPC framework for nonlinear systems. It is assumed that the linearization of the original nonlinear system is stabilizable at the origin. Then a local linear feedback law which stabilizes the linearization, can be determined. The linear feedback can also be proved to stabilize the original nonlinear system locally. Feasible control input to the optimization problem can be produced by the local control law and optimality is thus obtained. In this way, the stability is gained from the recursive feasibility and the optimality.

- To tackle additive disturbance caused by noise, model mismatch or parametric uncertainty, robust MPC schemes are developed. The existing literatures in robust MPC can be classified as three categories: robust MPC with nominal cost [44], tube-based MPC [45, 46] and min-max robust MPC [12, 47, 48]. Since MPC design combines the feedback mechanism and optimization, the inherent properties of feedback, to some extend, can provide a certain degree of robustness against external disturbance. By tightening the original constraints, satisfaction of restrictions on the actual disturbed systems can be achieved in robust MPC with nominal cost. This approach, however, generally yields conservative robustness in open-loop prediction [44]. A typical tube-based MPC scheme incorporates an a priori well-tuned linear feedback with control input generated from constrained optimization. The former static feedback, or referred to as nominal/reference feedback [45, 49], helps tackle effects from disturbance and the latter preserves constraint satisfaction. The conservativeness in [44] can be reduced, especially for nonlinear system dynamics, by using tube-based MPC strategies. In another type of tube-based MPC, also known as feedback MPC, the decision variable in the OCP is a policy, i.e., a sequence of control laws, rather than control actions (see [46, 50, 51] for examples). In min-max robust MPC, the worst case in all admissible disturbance is considered to guarantee the satisfactory of robust constraints. By using dynamic programming techniques, a min-max optimization problem is formulated and solved to generate control input. This approach provides better robustness but consumes more computational resources. Thus the trade-off between performance and computation has to be taken into consideration when implementing the min-max MPC.

A brief literature review for typical MPC schemes can be found in Table 1.2.

Table 1.2: Brief literature review for typical MPC schemes.

Related Work	**System Model**	**Deterministic /Robust**	**Input Generation**
Chen *et al*, 1998	Nonlinear & Continuous	Deterministic	Optimal control action
Mayne *et al*, 2000	Nonlinear & Discrete	Deterministic	Optimal control policy
Marruedo *et al*, 2002	Nonlinear & Continuous	Inherent robust	Optimal control action
Mayne *et al*, 2005; Chisci *et al*, 2001	Linear & Discrete	Tubed-based robust	Optimal control policy
Raimondo *et al*, 2009	Nonlinear & Discrete	Min-max robust	Optimal control policy

1.3.2 MPC-based Consensus

It is well known that MPC has proved success in handling hard constraints, and it is widely adopted in MAS control. Among MPC-based multi-agent control frameworks, most of the existing results discuss cooperative stabilization and tracking [52, 53, 54, 55], where the systems have to converge to an a priori known set point or to follow an a priori specified reference trajectory. Compared to the classical stabilization or tracking control problems, more general cooperative control objectives, in particular, multi-agent consensus, are of great importance. In such a setting, the systems are required to agree on a common online trajectory which is not necessarily a priori specified. MPC-based consensus is still challenging with relatively fewer results.

In the early stage, the main focus is on first-order integrator dynamics [54], and in recent years, some new approaches for double-integrator models appear (see [56, 23] for examples). In [23], a distributed model predictive controller is designed for static formation of a group of agents governed by double integrator dynamics. The cooperative control problem is formulated into an unconstrained leader-follower formation problem, and the leader agent has global access to the state of the overall system. The authors in [57] propose an MPC-based consensus strategy for unconstrained integrators. In this work, iterative information exchange among agents is required. Most existing works considering integrator models focus on fixed network topologies, but one exception studying MAS underlying a directed graph with switching topology can be found in [56].

In the literatures of consensus of linear MASs, some decent results are presented in [58]. In this work, the consensus problem of general linear MASs are investigated under the framework of unconstrained optimization. An explicit solution is derived by using Karush–Kuhn–Tucker (KKT) conditions and more specified consensus conditions for one-dimensional linear systems are developed by Riccati difference equations. However, input/state constraints are not involved in this framework. For constrained linear MASs, the researchers in [38] first propose an inverse optimal linear consensus protocol, such that the closed-loop system is asymptotically stable with respect to a consensus set. When the input constraints are involved, a centralized MPC-based consensus strategy is designed based on the pre-stabilizing linear consensus protocol. Distributed MPC scheme is later developed by decoupling the centralized one. Applications of distributed MPC-based consensus, like multi-vehicle platoon, can be found in [27]. This work proposes a novel distributed MPC-based consensus framework with self-triggered mechanism. Information accessing among agents happens only one time at every sampling instant and multi-vehicle platoon problem is later studied based on this distributed self-triggered MPC framework. However, the communication pattern in this control strategy requires simultaneous information sharing when each agent measures its current state in a directed network. This is particularly troublesome in networked control systems with distributed digital controller setup. Moreover, the conditions for recursive feasibility are not rigorously derived, thus a rather strong assumption is made in this work.

More related works on MPC-based consensus for different scenarios are listed in Table 1.3.

Table 1.3: Existing literatures for MPC-based consensus

Coupling	**Cost function**	**Constraints**
	Zhan *et al*, 2018	Müller *et al*, 2012a
Communication	**Iterative**	**Non-iterative**
	Müller *et al*, 2012b	Li *et al*, 2018
	Fixed topology	**Time-varying topology**
	Ferrari-Trecate *et al*, 2019	Cheng *et al*, 2015
	Periodic	**Triggered**
	Manfredi *et al*, 2018	Duan *et al*, 2018
Robustness	**Tube-based**	**Min-max**
	Müller *et al*, 2012b	Jia *et al*, 2002

1.4 Motivations and Contributions

1.4.1 Motivations

The main motivations of this thesis can be summarized in the following points of view.

- **Tube-based robust MPC consensus.** Many existing works on MPC-based consensus focus on simple deterministic system dynamics like single or double integrator systems [22, 23, 59], with several exceptions considering disturbance [12, 60] as mentioned in the previous section. In a notable result [60], where tube-based MPC is adopted to solve consensus problem of nonlinear systems, the model predictive controller assigned on each agent requires multiple times of communication to share both state and input information. This iterative communication pattern imposes heavy pressure on the communication network. Therefore, it would make perfect sense that the **tube-base robust** MPC consensus protocol designed for general **linear systems** with **disturbance** requires **non-iterative information sharing** among agents.

- **Distributed MPC-based consensus.** In the literature of MPC for multi-agent cooperation, most of the existing results focus on cooperative stabilization problem [32, 52, 39, 5]. Till now, there are still few results adopting distributed MPC for multi-agent consensus and some of them are reported in [54, 61, 58, 38, 27]. In [54], the system model is restricted to integrator dynamics and is not easy to extend to more general system models. Distributed MPC frameworks involving more general constrained linear systems can be found in [27, 38], but simultaneous access to neighboring agent state is required. This is especially difficult to realize in digital controller networks. The authors in [57, 58] study multi-agent consensus by formulating the control problem into an optimization problem without considering constraints. None of the existing MPC results can handle multi-agent consensus for general **constrained linear** MASs underlying an undirected communication graph with **truly distributed** local controllers.

1.4.2 Contributions

This thesis focuses on designing MPC-based consensus control strategies for different system and network setups. The main contributions of this thesis are summarized as follows.

- **Centralized robust MPC-based consensus for linear MASs with input constraints.** In the first part, we propose the tube-based MPC scheme to solve the consensus problem of constrained linear MASs with bounded additive disturbance. A linear consensus protocol is first introduced for the undisturbed systems without constraints. Due to the non-convexity of the infinite horizon performance function in linear quadratic form, the optimal controller gain is difficult, or even impossible to find. Thus a suboptimal solution is adopted and the sufficient and necessary conditions for its existence are given. By solving a modified discrete-time Riccati equation (DARE), an identical local feedback gain is obtained for every agent in the MAS. When disturbance and constraints are involved, a tube-based MPC consensus strategy is adopted. By inserting tightened constraints into the original OCP, the closed-loop systems gain robustness against disturbance. We further prove that the recursive feasibility and convergence can be guaranteed.

- **Distributed MPC-based consensus for linear systems with input constraints.** The second part of this thesis is concerned with a novel distributed MPC-based consensus protocol for semi-stable systems with input saturation. At every time step, each agent measures its current state, then solves a local constrained OCP to generate control signals, where only state information of the neighbors is required. Once the local optimization problem is solved, every agent exchanges the predicted state information with its neighbors only once at each time instant, i.e., no iterative communication is required. We first introduce a linear consensus protocol designed via the inverse optimal method as the pre-stabilizing control law, such that the closed-loop system is asymptotically stable with respect to a consensus set. When input saturation is considered, we minimize the gap between the MPC input and the pre-stabilizing controller input while preserving the satisfaction of input constraints. Moreover, an additional state constraint is inserted into the OCP in order to restrict the deviation between the acutal and the assumed state. By doing this, cooperation among agents is reinforced. We further prove the feasibility and show that the MAS converges to a neighboring set of the consensus set.

1.5 Thesis Organization

The reminder of this thesis is organized as follows:

Chapter 2 involves a centralized robust MPC scheme to solve the consensus problem of general linear multi-agent systems with bounded disturbance.

Chapter 3 proposes a novel distributed MPC scheme to solve the consensus problem for general semi-stable linear MASs in truly distributed networks.

Chapter 4 gives the conclusions of this thesis and proposes some interesting future research areas.

Chapter 2

A Centralized Robust MPC-based Consensus Protocol for Disturbed Multi-agent Systems

2.1 Introduction

In this chapter, our focus is on robust model predictive solution to consensus problem of linear MASs subject to persistent additive disturbance. We first present the results on developing a centralized linear consensus protocol, via suboptimal linear quadratic approach, for unconstrained nominal MASs. When disturbance and constraints are considered in the MPC framework, the original constraint sets in nominal MPC prediction should be more stringent, so that the actual state/input, which does not necessarily coincide with the predicted ones, can achieve robust constraint satisfaction. We later show the constraint sets in our MPC framework can be offline determined and no extra online computation is required. Properties such as stability and iterative feasibility will be provided.

In applications involving multiple agents agreeing upon various quantitative interests, consensus has been a long-standing area of research. The past few decades have witnessed an enormous amount of research efforts on cooperative control of MASs from different perspectives. In the early stage, the main focus is on consensus of multiple agents governed by first-order dynamics [62, 18, 19], where necessary and sufficient conditions for first-order dynamics under different setups are illustrated. The interest in studying consensus of second-order dynamics also grows in the past decade, and some decent results can be found in [24, 5]. Particularly, the authors in [24] demonstrate necessary and sufficient conditions for MASs governing by second-order dynamics underlying a fixed connected information transmission network topology.

In recent years, more and more results are proposed to handle practical issues in MAS consensus. Many of these works involve practical constraints under different network setups, including actuator saturation [25], collision avoidance [26, 31], time-delay [63, 64] and switching network topology [19, 65]. These techniques turn out to be useful in industrial applications. For example, the authors in [6] consider a multiple-robot system with physically decoupled nonlinear dynamics, pursuing a common cooperative control task subject to certain coupling constraints.

It is well known that MPC has been broadly implemented in many industrial applications for decades and has significant profits in handling hard constraints in comparison with many other conventional control strategies [66]. In the literature, many works consider deterministic linear/nonlinear systems and illustrate conditions ensuring stability. Some of those remarkable results can be found in [67, 68, 69, 43].

However, the presence of uncertainty, in possible form of additive disturbance, inaccurate state estimation or model mismatch, may destroy stability of nominal predictive control systems. To attenuate the uncertainty effect, the authors in [44] investigate the growth of the disturbance effect on nominal systems along the prediction horizon and introduce the robust MPC with nominal cost. This framework benefits from some degree of inherent robustness of MPC. By involving properly tightened constraint sets in nominal prediction, the uncertainty effect can be limited. This approach, however, may bring in conservatism because of its open-loop fashion in prediction. Another approach considers deviation between the predicted state and the actual state. The deviation is then characterized by a sequence of limited sets, also known as “tubes”. By subtracting these deviation sets from the original constraint sets, the tightened constraints for robust prediction are obtained. After solving the robust OCP, a sequence of control actions is generated. This category of tube-based MPC cannot contain the “spread” of predicted state/input trajectory, so this prediction is open-loop and may result in conservatism. In another type of tube-based MPC scheme, also known as feedback MPC, see [46, 50, 51] for examples, the decision variable in the optimal control problem is a policy, or namely a sequence of control laws, rather than a sequence of control actions. Since disturbance considered in robust MPC problems is assumed to be bounded, another well-known feedback MPC framework, min-max MPC, involves the worst case of all possible disturbance to satisfy constraints and solves a min-max optimization problem to obtain a sequence of optimal control policies [47]. However, the min-max MPC is the most computationally expensive among these three categories of robust MPC frameworks. Supplementary techniques like parameterization of policies are developed to reduce the degree of freedom of the min-max optimization problem, so that computation load is reduced to a practically solvable level [48].

Compared with the robust MPC with nominal cost, the tube-based MPC has more profits in handling disturbance and yield less conservative robustness margins. It also consumes less computation power than the min-max MPC frameworks, so the required computation resource level is more practically acceptable. Motivated by this fact, we want to investigate the consensus problem in disturbed linear MASs by making use of the tube-based MPC.

The main contributions of this chapter are two-fold:

- We present a linear consensus protocol design method for linear MASs by extending the results in [28] into discrete-time domain. Sufficient and necessary conditions for the existence of such a suboptimal solution to the linear quadratic consensus control problem are given. By computing a positive definite solution of a modified discrete-time algebraic Riccati equation (DARE), an identical local controller gain is obtained for every agent.

- A tube-based MPC scheme is developed to tackle infeasibility and instability of a predictive controller when joint presence of input constraints and additive disturbance occurs. The robustness is enforced by inserting restricted constraints into the nominal predictive controller, so that input-to-state stability is guaranteed.

The reminder of this chapter is organized as follows. Section 2.2 formulates the robust consensus problem and presents the control objectives. Section 2.3 develops the consensus protocol designed via the suboptimal linear quadratic approach. With the linear consensus protocol, we introduce a nominal predictive consensus control and show the offline computed constraint sets of the MPC framework at the beginning of Section 2.4, followed by the robust model predictive solution to the consensus problem. Section 2.5 focuses on feasibility and convergence analysis. Numerical examples and simulation study are provided in Section 2.6. Section 2.7 concludes this chapter.

Notation: The notation $\mathbb{R}$ represents the set of real numbers and $\mathbb{R}^n$ denotes the Cartesian product of $\underbrace{\mathbb{R} \times \cdots \times \mathbb{R}}_{n}$. A sequence of integers is given by $\mathbb{N}_{[m,n]} = \{m, m+1, \ldots, n\}$. Given two sets A, $B \subseteq \mathbb{R}^n$ and vectors $a, b, c \in \mathbb{R}^n$, the Minkowski sum of the sets is defined by $A \oplus B := \{c | c = a + b, a \in A, b \in B\}$ and the Pontryagin difference of the two sets is given by $A \ominus B := \{c | c + b \in A, b \in B\}$. The Euclidean norm is denoted by $\|\cdot\|$ and for a given matrix P, the weighted norm of a vector $x \in \mathbb{R}^n$ is defined by $\|x\|_P^2 = x^{\mathrm{T}} P x$. Let $S(r) = \{\boldsymbol{x}_0 \in \mathbb{R}^{nM} | \, \|\boldsymbol{x}_0\|^2 \leq r^2\}$ be the closed sphere of radius r in the nM-dimensional space.

2.2 Preliminaries and Problem Statement

2.2.1 Preliminaries

In this chapter, we consider an MAS consisting of M identical agents of the form

$$x_i(k+1) = Ax_i(k) + Bu_i(k) + \omega_i(k), \quad i \in \mathbb{N}_{[1,M]}, \tag{2.1}$$

with $x_i \in \mathbb{R}^n$ being the state and $u_i \in \mathcal{U}_i \subseteq \mathbb{R}^m$ being the control input. An unknown disturbance acting on agent i is denoted by $\omega_i \in \mathcal{W}_i \subseteq \mathbb{R}^n$. By using the Kronecker product, we can rewrite the MAS of the dynamics in (2.1) in compact form as

$$\boldsymbol{x}(k+1) = (I_M \otimes A)\boldsymbol{x}(k) + (I_M \otimes B)\boldsymbol{u}(k) + \boldsymbol{\omega}(k), \tag{2.2}$$

with $\boldsymbol{x} = \begin{bmatrix} x_1^{\mathrm{T}}, \cdots, x_M^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$, $\boldsymbol{u} = \begin{bmatrix} u_1^{\mathrm{T}}, \cdots, u_M^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ and $\boldsymbol{\omega} = \begin{bmatrix} \omega_1^{\mathrm{T}}, \cdots, \omega_M^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$. The vectors $\boldsymbol{x} \in \mathbb{R}^{n \times M}$, $\boldsymbol{u} \in \mathbb{R}^{m \times M}$ and $\boldsymbol{\omega} \in \mathbb{R}^{n \times M}$ denote the state, the control input and the disturbance of the overall system. We also denote the input constraint set and disturbance set by $\mathcal{U} := \mathcal{U}_1 \times \cdots \times \mathcal{U}_M$ and $\mathcal{W} := \mathcal{W}_1 \times \cdots \times \mathcal{W}_M$, respectively.

Graph theory is one of the most commonly used mathematical tools in modelling information exchange for MASs. In this chapter, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denotes a simple undirected graph, where $\mathcal{V} = \{v_1, v_2, \cdots, v_M\}$ represents the vertex set and $\mathcal{E} = \{(v_i, v_j) \, | v_i, v_j \in \mathcal{V}, i \neq j\} \subset \mathcal{V} \times \mathcal{V}$ represents the edge set. Let $\mathcal{N}_i$ be the set of all neighboring vertices of node i, i.e., $\mathcal{N}_i := \{v_j | v_i, v_j \in \mathcal{V}, (v_i, v_j) \in \mathcal{E}, i \neq j\}$ and $d_i := |\mathcal{N}_i|$ its cardinality. For a graph $\mathcal{G}$ with M vertices, its adjacency matrix $\mathcal{A}^d = [a_{ij}] \in \mathbb{R}^{M \times M}$ is given by $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$ and by $a_{ij} = 0$ if $(i, j) \notin \mathcal{E}$. Accordingly, the Laplacian matrix is defined as $\mathcal{L} = [l_{ij}] \in \mathbb{R}^{n \times n}$, where

$$\begin{cases} l_{ij} = -a_{ij}, \quad \forall i \neq j \\ l_{ii} = \sum_{j \in \mathcal{N}_i} a_{ij}, \quad i, j = \mathbb{N}_{[1,M]}. \end{cases}$$

The Laplacian matrix of an undirected graph has the following properties. It is symmetric and has no negative eigenvalues which can be ordered as $\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_M$ with $\lambda_1 = 0$. Therefore, we can always find an orthogonal matrix $U \in \mathbb{R}^{M \times M}$, such that $U^{\mathrm{T}} \mathcal{L} U = \mathrm{diag}\{\lambda_1, \lambda_2, \cdots, \lambda_M\}$, where U^{T} is the transpose of matrix U. We also define the diagonal matrix as $\Lambda := U^{\mathrm{T}} \mathcal{L} U = \mathrm{diag}\{0, \lambda_2, \cdots, \lambda_M\}$.

Definition 1. *A path from vertex i_1 to i_k is denoted by an edge sequence*

$$\{(i_1, i_2), (i_2, i_3) \cdots (i_{k-1}, i_k)\},$$

with all edges in the sequence (i_{j-1}, i_j) or (i_j, i_{j-1}), $j \in \mathbb{N}_{[1,k]}$ belonging to the edge set $\mathcal{E}$. If there exists a vertex i such that any other vertices in graph $\mathcal{G}$ can be reached via at least one path, the graph $\mathcal{G}$ is said to contain a spinning tree and to be connected.

Definition 2. *For the linear MAS in (2.1) over a connected undirected graph $\mathcal{G}$, it is said to reach consensus if*

$$\|x_i(k) - x_j(k)\| \to 0 \text{ as } k \to \infty,$$

where $\forall i, j \in \mathbb{N}_{[1,M]}, i \neq j$.

The assumptions providing necessary conditions for the MAS in (2.1) achieving consensus are given as follows.

Assumption 1. *In the reminder of the chaper, we assume that*

1. *The pair (A, B) is assumed to be stabilizable.*

2. *The input constraint set $\mathcal{U}_i$ and the disturbance set $\mathcal{W}_i$ are compact and contain the origin as their interior point.*

3. *The graph $\mathcal{G}$ is connected and contains a spanning tree, giving a necessary condition to achieve consensus.*

2.2.2 Control Objective

Our objective is to design a nonlinear consensus feedback

$$\boldsymbol{u}(k) = g(\boldsymbol{x}(k)), \tag{2.3}$$

via MPC, which regulates the MAS in (2.2) to reach consensus while satisfying the constraints for all possible disturbance. The continuous nonlinear function $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$. Furthermore, the nonlinear consensus protocol in (2.3) reduces to an a priori well-tuned linear consensus feedback when the system state enters a target set $\mathcal{X}^f$ (to be specified later). Due to the persistent disturbance acting on each agent, the MAS in (2.2) is not possible to achieve consensus asymptotically. Then our best hope would be to steer the state disagreement $\sum_{i,j\in\mathcal{V}} \|x_i(k) - x_j(k)\|$ as small as possible.

To meet the desire for the well-tuned consensus control when $\boldsymbol{x}(k) \in \mathcal{X}^f$, we consider a consensus protocol of the form

$$u_i^l(k) = K \sum_{j\in\mathcal{N}_i} (x_i(k) - x_j(k)),$$

or namely,

$$\boldsymbol{u}^l(k) = (\mathcal{L} \otimes K)\boldsymbol{x}(k), \tag{2.4}$$

so the corresponding unperturbed closed-loop system becomes

$$\begin{aligned} \boldsymbol{x}(k+1) &= (I_M \otimes A)\boldsymbol{x}(k) + (I_M \otimes B)\boldsymbol{u}^l(k) \\ &= (I_M \otimes A + \mathcal{L} \otimes BK)\boldsymbol{x}(k) := \Phi\boldsymbol{x}(k), \end{aligned} \tag{2.5}$$

where $\Phi := I_M \otimes A + \mathcal{L} \otimes BK$ is the state transition matrix and K is the feedback gain matrix (to be later specified). To collaborate the linear consensus feedback with the MPC-based consensus framework, we introduce the control variable,

$$\boldsymbol{c}(k) := \boldsymbol{u}(k) - \boldsymbol{u}^l(k) = \boldsymbol{u}(k) - (\mathcal{L} \otimes K)\boldsymbol{x}(k), \tag{2.6}$$

to characterize the difference between the MPC input and the well-tuned linear consensus control input. Accordingly, the disturbed MAS in (2.2) can be rewritten as

$$\boldsymbol{x}(k+1) = \Phi\boldsymbol{x}(k) + (I_M \otimes B)\boldsymbol{c}(k) + \boldsymbol{\omega}(k). \tag{2.7}$$

Now we can restate the control objective in this chapter as follows:

- Design a consensus protocol given by (2.3) using MPC framework for the disturbed MAS in (2.2).
- A further requirement is to vanish the gap between the MPC input and the nominal control input: $\lim\limits_{k\to\infty} \boldsymbol{c}(k) = 0$.

In the following section, we demonstrate how to design the linear consensus protocol by using a suboptimal method.

2.3 Suboptimal Consensus Protocol Design

In this section, we discuss the consensus problem for linear MASs by extending [28] into discrete-time systems. Consider an MAS represented by

$$x_i(k+1) = Ax_i(k) + Bu_i(k), \quad x_i(0) = x_{i0}, \quad i \in \mathbb{N}_{[1,M]}, \tag{2.8}$$

where matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ and the system state and the control input of agent i are denoted by $x_i \in \mathbb{R}^n$ and $u_i \in \mathbb{R}^m$, respectively. To evaluate the consensus performance, an infinite horizon linear quadratic cost function is implemented:

$$J_i(x_i, x_j, u_i) = \frac{1}{2}\sum_{k=0}^{\infty}\sum_{j\in\mathcal{N}_i} \|x_i(k) - x_j(k)\|_Q^2 + \|u_i(k)\|_R^2\,, \tag{2.9}$$

where $Q \succeq 0$, $R \succ 0$ are real weighting matrices. This performance function sums the weighted norm of the state disagreement among every agent and its neighbors, and it also penalizes the control input in quadratic form.

Concatenating the states and the inputs of all agents in columns, we can write (2.8) in compact form as

$$\boldsymbol{x}(k+1) = (I_M \otimes A)\boldsymbol{x}(k) + (I_M \otimes B)\boldsymbol{u}(k), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0. \tag{2.10}$$

The initial states of the agents are collected in the joint column vector to represent the overall system initial state $\boldsymbol{x}_0$. Accordingly, the performance functional of the overall MAS in (2.10) can be written as

$$J(\boldsymbol{x}, \boldsymbol{u}) = \sum_{i=1}^{M} J_i(x_i, x_j, u_i) = \sum_{k=0}^{\infty} \|\boldsymbol{x}(k)\|_{(\mathcal{L}\otimes Q)}^2 + \|\boldsymbol{u}(k)\|_{(I_M\otimes R)}^2\,. \tag{2.11}$$

We want to find a linear state feedback of the form

$$u_i(k) = K \sum_{j \in \mathcal{N}_i} (x_i(k) - x_j(k)), \tag{2.12}$$

or namely,

$$\boldsymbol{u}(k) = (\mathcal{L} \otimes K)\boldsymbol{x}(k). \tag{2.13}$$

where $K \in \mathbb{R}^{n\times m}$ is an identical feedback gain for every agent, such that the performance functional in (2.11) is minimized. Accordingly, the closed-loop system is

$$\boldsymbol{x}(k+1) = (I_M \otimes A + \mathcal{L} \otimes BK)\boldsymbol{x}(k). \tag{2.14}$$

The associated performance function in (2.11) can be written as a function of the gain matrix K:

$$\begin{aligned} J(K) &= J(\boldsymbol{x}, \boldsymbol{u}) = J(\boldsymbol{x}, (\mathcal{L} \otimes K)\boldsymbol{x}) \\ &= \sum_{k=0}^{\infty} \|\boldsymbol{x}(k)\|^2_{(\mathcal{L}\otimes Q)} + \|(\mathcal{L} \otimes K)\boldsymbol{x}(k)\|^2_{(I_M \otimes R)} \\ &= \sum_{k=0}^{\infty} \boldsymbol{x}^{\mathrm{T}}(k)(\mathcal{L} \otimes Q + \mathcal{L}^{\mathrm{T}}\mathcal{L} \otimes K^{\mathrm{T}}RK)\boldsymbol{x}(k) \\ &= \sum_{k=0}^{\infty} \|\boldsymbol{x}(k)\|^2_{(\mathcal{L}\otimes Q + \mathcal{L}^{\mathrm{T}}\mathcal{L}\otimes K^{\mathrm{T}}RK)} . \end{aligned} \tag{2.15}$$

Due to the nature of an undirected graph, it always holds that the minimum eigenvalue of the Laplacian is zero, i.e., $\lambda_{\min}(\mathcal{L}) = 0$. Thus, the weighting matrix of the infinite horizon objective function in (2.15) is not guaranteed to be positive definite, or namely,

$$\mathcal{L} \otimes Q + \mathcal{L}^{\mathrm{T}}\mathcal{L} \otimes K^{\mathrm{T}}RK \succeq 0.$$

This indicates that the corresponding minimization problem is non-convex. It is difficult or even impossible to find an optimal consensus feedback gain K in such case, or the optimal solution may not even exist. We will instead, solve the consensus problem by involving the suboptimal solution. More precisely, the following problem is considered:

Problem 1. *Consider the MAS in (2.8) over an undirected graph $\mathcal{G}$ with the initial state $\boldsymbol{x}(0) = \boldsymbol{x}_0$. Denote an a priori known upper bound for the performance function (2.15) by a positive constant ρ. We want to find a consensus protocol given by (2.12) for each agent, so that the closed-loop system in (2.14) achieves consensus and the associated performance function (2.15) is less than the upper bound ρ.*

Following the same line in solving the discrete-time algebraic Riccati equation (DARE) in linear quadratic regulator (LQR) design, we first address the suboptimal control problem for a single agent.

2.3.1 Suboptimal Solution to Autonomous Systems

In this subsection, we analyze the suboptimal solution to an autonomous system. Consider an autonomous system given by

$$x(k+1) = \bar{A}x(k), \quad x(0) = x_0, \tag{2.16}$$

where $\bar{A} \in \mathbb{R}^{n\times n}$ and $x \in \mathbb{R}^n$ is the system state. The infinite horizon quadratic performance function associated with (2.16) is given by

$$J(x) := \sum_{k=0}^{\infty} \|x(k)\|_{\bar{Q}}^2 = \sum_{k=0}^{\infty} x(k)^{\mathrm{T}} \bar{Q} x(k), \tag{2.17}$$

where $\bar{Q} \succeq 0$ is a given weighting matrix.

Lemma 1 (Theorem B.18 in [70]). *The performance function (2.17) gives a finite value if the state transition matrix of the corresponding autonomous system in (2.16) is Schur stable, i.e., the eigenvalues of $\bar{A}$ are in the unit circle. In this case, the infinite horizon performance function (2.17) gives a finite value*

$$J = x_0^{\mathrm{T}} Y x_0, \tag{2.18}$$

where the unique matrix $Y \succeq 0$ is the solution to

$$\bar{A}^{\mathrm{T}} Y \bar{A} - Y + \bar{Q} = 0. \tag{2.19}$$

By the properties of the discrete-time Lyapunov equation, it is a well known fact that the quadratic performance (2.17) is given by the weighted norm of the initial state (2.18), so we just omit the proof here. Alternatively, we are more interested in finding the solution given by a series of Lyapunov inequalities, so that the corresponding performance function (2.17) is less than a given constant $\rho > 0$.

Lemma 2. *Consider the system in (2.16) with the corresponding linear quadratic performance function (2.17). If the system is Schur stable, the expression of the performance function given by the Lyapunov inequality,*

$$\inf\{x_0^{\mathrm{T}} P x_0 | P \succeq 0, \bar{A}^{\mathrm{T}} P \bar{A} - P + \bar{Q} < 0\}, \tag{2.20}$$

is equivalent to (2.18).

Proof. Suppose that Y is the solution to Lyapunov equation in (2.19) and $P \succeq 0$ is the solution to Lyapunov inequality in (2.20). Let $X := P - Y$, one gets

$$\begin{aligned} \bar{A}^{\mathrm{T}}(X+Y)\bar{A} - (X+Y) + \bar{Q} &< 0 \\ \bar{A}^{\mathrm{T}}X\bar{A} + \bar{A}^{\mathrm{T}}Y\bar{A} - (X+Y) + \bar{Q} &< 0 \\ \bar{A}^{\mathrm{T}}X\bar{A} &< X. \end{aligned} \tag{2.21}$$

Since $\bar{A}$ is Schur stable, we immediately have $X \succ 0$, or namely, $P - Y \succ 0$. Consequently, it holds that

$$J = x_0^{\mathrm{T}} Y x_0 \leq x_0^{\mathrm{T}} P x_0, \tag{2.22}$$

for any positive semi-definite solution P to the Lyapunov inequality. Therefore, the infimum expression (2.20) is exactly (2.18):

$$J = \inf\{x_0^{\mathrm{T}} P x_0 | P \succeq 0 \text{ and } \bar{A}^{\mathrm{T}} P \bar{A} - P + \bar{Q} < 0\} = x_0^{\mathrm{T}} Y x_0.$$

□

Remark 1. *In fact, one can always find a positive semi-definite matrix P_ϵ satisfying Lyapunov inequality in (2.20) with $P_\epsilon < Y + \epsilon I$, for any given $\epsilon > 0$. Obviously, matrix P_ϵ can be chosen by solving the following Lyapunov equality*

$$\bar{A}^{\mathrm{T}} P_\epsilon \bar{A} - P_\epsilon + \bar{Q} + \epsilon I = 0.$$

It can also be noticed that $P_\epsilon \to Y$ as $\epsilon \to 0$. Therefore, for any given upper bound, we can always find a suboptimal solution P_ϵ satisfying $J \leq x_0^{\mathrm{T}} P_\epsilon x_0$, by solving the corresponding Lyapunov equation.

To evaluate under what conditions the performance function (2.17) is smaller than a given upper bound, the following theorem is proposed:

Theorem 1. *Consider the autonomous system in (2.16) with the associated positive semi-definite performance function (2.17). For a given constant $\rho > 0$, it holds that $\bar{A}$ is Schur stable and $J < \rho$ iff a positive semi-definite P to the inequalities*

$$\bar{A}^{\mathrm{T}} P \bar{A} - P + \bar{Q} < 0, \tag{2.23}$$

and

$$x_0^{\mathrm{T}} P x_0 < \rho, \tag{2.24}$$

can be found.

Proof. (if) By Lemma 2, if there exists a positive semi-definite solution to (2.23), it holds that $\bar{A}$ is Schur stable. Since matrix P satisfies both (2.23) and (2.24), by taking a proper ϵ as in Remark 1, we can immediately have

$$J < x_0^{\mathrm{T}} P x_0 < \rho. \tag{2.25}$$

(only if) By Lemma 2 again, If $\bar{A}$ is Schur stable and $J < \rho$, then there exists a positive semi-definite solution P to (2.23), satisfying $J < x_0^{\mathrm{T}} P x_0 < \rho$. □

2.3.2 Suboptimal Solution to General Linear Systems

In this subsection, we analyze the suboptimal solution to general linear discrete systems with control input. Consider a discrete-time LTI system given by

$$x(k+1) = Ax(k) + Bu(k), \quad x(0) = x_0, \tag{2.26}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{m\times m}$. The system state is $x \in \mathbb{R}^n$ and the control input is $u \in \mathbb{R}^m$, and the associated linear quadratic performance function is given by

$$J(x,u) = \sum_{k=0}^{\infty} \|x(k)\|_Q^2 + \|u(k)\|_R^2 \tag{2.27}$$

where the weighting matrices $Q \succeq 0$ and $R \succ 0$. It is assumed that the pair (A, B) is stabilizable. We want to find a linear state feedback $u = Kx$, such that the associated closed-loop system

$$x(k+1) = (A + BK)x(k) \tag{2.28}$$

is asymptotically stable and the corresponding cost functional (2.27),

$$J(x,u) = \sum_{k=0}^{\infty} \|x(k)\|_Q^2 + \|Kx(k)\|_R^2 = \sum_{k=0}^{\infty} \|x(k)\|^2_{(Q+K^{\mathrm{T}}RK)},$$

which can be regarded as a function of the feedback matrix K,

$$J(K) = \sum_{k=0}^{\infty} \|x(k)\|^2_{(Q+K^{\mathrm{T}}RK)},$$

is less than a given upper bound ρ.

The following lemma provides a sufficient condition for the existence of such a feedback.

Lemma 3. *If there exists a matrix $P \succeq 0$, such that*

$$A^{\mathrm{T}}PA - A^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA + Q - P < 0, \tag{2.29}$$

$$x_0^{\mathrm{T}}Px_0 < \rho, \tag{2.30}$$

then the linear feedback $u(k) = Kx(k)$, where the feedback gain matrix is given by

$$K = -(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA, \tag{2.31}$$

can stabilize the closed-loop system in (2.28) and the associated performance function (2.27) is bounded by ρ.

Proof. With the linear feedback given by (2.31), the closed-loop system in (2.28) becomes

$$x(k+1) = (A - B(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA)x(k), \quad x(0) = x_0. \tag{2.32}$$

Let

$$\bar{A} := A - B(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA$$

and

$$\begin{aligned}\bar{Q} :=& Q + A^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}BPA \\ &- A^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA.\end{aligned}$$

By taking

$$Z := (R + B^{\mathrm{T}}PB)^{-1} - (R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}$$

and multiplying a symmetric term $(R + B^{\mathrm{T}}PB) \succeq 0$ on both sides of Z, it holds that

$$(R + B^{\mathrm{T}}PB)Z(R + B^{\mathrm{T}}PB) = (R + B^{\mathrm{T}}PB) - B^{\mathrm{T}}PB = R \succ 0,$$

so we can check that $Z \succeq 0$. Since $\bar{Q} = Q + A^{\mathrm{T}}PBZB^{\mathrm{T}}PA$, it can be easily verified that $\bar{Q} \succeq 0$. One can evaluate the corresponding Lyapunov inequality as follow:

$$
\begin{aligned}
&\bar{A}^{\mathrm{T}} P \bar{A} - P + \bar{Q} \\
=&(A - B(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA)^{\mathrm{T}} P (A - B(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA) - P + \bar{Q} \\
=&A^{\mathrm{T}}PA - 2A^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA \\
&+ A^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA - P + \bar{Q} \\
=&A^{\mathrm{T}}PA - A^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA + Q - P.
\end{aligned}
$$

Since matrix P satisfies (2.29), we immediately have

$$
\bar{A}^{\mathrm{T}} P \bar{A} - P + \bar{Q} = A^{\mathrm{T}}PA - A^{\mathrm{T}}PB(R + B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA + Q - P < 0. \tag{2.33}
$$

By Lemma 2 and Theorem 1, the closed-loop system in (2.32) is asymptotically stable and the corresponding cost function $J < \rho$. $\square$

Remark 2. *By implementing Lemma* 3*, we can find a group of suboptimal linear feedbacks satisfying $J < \rho$, as long as there exists any matrix P satisfying (2.29) and (2.30).*

In the following subsection, we will apply the proposed design method to solve linear quadratic consensus control problem for a linear MAS.

2.3.3 Subpotimal Solution to Multi-agent Systems

As stated in the beginning of this section, we would like to find a linear control law given by (2.12) that regulates an MAS in (2.8) to achieve consensus. In the meanwhile, the corresponding performance function (2.15) is less than a given upper bound ρ.

We apply the state and input transformations

$$\begin{aligned} \bar{\boldsymbol{x}} &= \begin{bmatrix} \bar{x}_1^{\mathrm{T}} & \bar{x}_2^{\mathrm{T}} & \cdots & \bar{x}_M^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} = (U^{\mathrm{T}} \otimes I_n)\boldsymbol{x}, \\ \bar{\boldsymbol{u}} &= \begin{bmatrix} \bar{u}_1^{\mathrm{T}} & \bar{u}_2^{\mathrm{T}} & \cdots & \bar{u}_M^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} = (U^{\mathrm{T}} \otimes I_m)\boldsymbol{u}, \end{aligned} \tag{2.34}$$

or namely,

$$\boldsymbol{x} = (U^{-\mathrm{T}} \otimes I_n)\bar{\boldsymbol{x}}, \quad \boldsymbol{u} = (U^{-\mathrm{T}} \otimes I_m)\bar{\boldsymbol{u}}, \tag{2.35}$$

to decouple the MAS, where matrix U satisfies $U^{\mathrm{T}}\mathcal{L}U = \Lambda = \operatorname{diag}\{0, \lambda_2, \cdots, \lambda_M\}$ and $U^{-\mathrm{T}}$ denotes the inverse of U^{T}. The MAS in (2.10) becomes

$$\bar{\boldsymbol{x}}(k+1) = (I_M \otimes A)\bar{\boldsymbol{x}}(k) + (I_M \otimes B)\bar{\boldsymbol{u}}(k), \quad \bar{\boldsymbol{x}}_0 = (U^{\mathrm{T}} \otimes I_n)\boldsymbol{x}_0. \tag{2.36}$$

Accordingly, the consensus protocol (2.13) for the system in (2.36) can be written as

$$\bar{\boldsymbol{u}}(k) = (\Lambda \otimes K)\bar{\boldsymbol{x}}(k). \tag{2.37}$$

Note that the transformed state $\bar{x}_i$ and input $\bar{u}_i$ of agent i are decoupled from its neighbors' states and inputs, so the linear feedback control (2.12) becomes

$$\begin{cases} \bar{u}_1(k) = 0, \\ \bar{u}_i(k) = \lambda_i K \bar{x}_i(k), \quad i \in \mathbb{N}_{[2,M]}. \end{cases} \tag{2.38}$$

The transformed dynamics of the agents can be decoupled into M autonomous systems given by

$$\bar{x}_1(k+1) = A\bar{x}_1, \tag{2.39}$$

$$\bar{x}_i(k+1) = (A + \lambda_i BK)\bar{x}_i(k), \tag{2.40}$$

where $i \in \mathbb{N}_{[2,M]}$. The performance function associated with the transformed system in (2.36) becomes

$$J(K) = \sum_{i=0}^{M} J_i(K) = \sum_{i=0}^{M}\sum_{k=0}^{\infty} \bar{x}_i^{\mathrm{T}}(k)(\lambda_i Q + \lambda_i^2 K^{\mathrm{T}} RK)\bar{x}_i(k). \tag{2.41}$$

It is worth mentioning that the first agent in (2.39) does not contribute to the performance function (2.41) as $\lambda_1 = \lambda_{\min}(\mathcal{L}) = 0$. The original overall system in (2.2) reaches consensus if and only if the decoupled systems in (2.40) are stabilized by the consensus control (2.37) with proper feedback gains. The following lemma givens sufficient condition for the existence of the suboptimal control feedback.

Lemma 4. *We consider system dynamics given by (2.36) with the corresponding cost function (2.41). For any given upper bound $\rho > 0$ and for all $\boldsymbol{x}_0 \in S(r)$ with a radius of r, all systems are stable and the cost function $J(K)$ is bounded by ρ, if positive semi-definite matrices P_i exist, such that*

$$(A + \lambda_i BK)^{\mathrm{T}} P_i (A + \lambda_i BK) - P_i + (\lambda_i Q_i + \lambda_i^2 K^{\mathrm{T}} RK) < 0, \tag{2.42}$$

$$\sum_{i=2}^{M} \bar{x}_{i0}^{\mathrm{T}} P_i \bar{x}_{i0} < \rho_i, \tag{2.43}$$

where $\sum_{i=1}^{M} \rho_i \le \rho$.

Proof. By Lemma 3, we can always find sufficient small ϵ_i associating with matrices P_i for each agent i, by taking $\bar{A}_i = A + \lambda_i BK$ and $\bar{Q}_i = \lambda_i Q_i + \lambda_i^2 K^{\mathrm{T}} RK$, such that $\bar{A}_i$ is Schur stable and $J_i \le \bar{x}_{i0} P_i \bar{x}_{i0} < \rho_i$. Since ϵ_i can be taken sufficiently small, we always have

$$J(K) = \sum_{i=1}^{M} J_i(K) \le \sum_{i=1}^{M} \bar{x}_{i0}^{\mathrm{T}} P_i \bar{x}_{i0} < \sum_{i=1}^{M} \rho_i \le \rho. \tag{2.44}$$

□

Remark 3. *Lemma 4 provides a series of control feedback gains to the transformed system dynamics. By taking inverse transformation, we immediately obtain the sub-optimal solution,*

$$u_i(k) = -(R + B^{\mathrm{T}} P_i B)^{-1} B^{\mathrm{T}} P_i A x_i(k), \tag{2.45}$$

to the consensus control problem, together with proper selected P_i obtained by Lemma 4.

By applying Lemma (4), the obtained linear consensus feedback gains for different agents may be different. In order to obtain an identical suboptimal feedback gain matrix satisfying all Lyapunov inequalities for all agents, the following lemma is proposed.

Lemma 5 (Theorem 9 in [28])**.** *For the suboptimal consensus control problem for the MAS in (2.10), the consensus protocol (2.13), whose gain is given by*

$$K = -c(R + B^{\mathrm{T}} P B)^{-1} B^{\mathrm{T}} P A, \quad c = \frac{2}{\lambda_2 + \lambda_M}, \tag{2.46}$$

is a common solution to all agents, where P is the solution to the following modified Lyapunov inequality

$$A^{\mathrm{T}} P A - (c^2 \lambda_M^2 - 2c\lambda_M) A^{\mathrm{T}} P B (R + B^{\mathrm{T}} P B)^{-1} B^{\mathrm{T}} P A + \lambda_M Q - P < 0,$$

$$\boldsymbol{x}_0^{\mathrm{T}} \{ (I_M - \frac{1}{N} \mathbf{1}^{\mathrm{T}} \mathbf{1}) \otimes P \} \boldsymbol{x}_0 < \rho,$$

Note that λ_2 and λ_M are the eigenvalues of the Laplacian matrix underlying the graph *of the communication network. This claim holds for all* $\boldsymbol{x}_0 \in S(r)$ *with a radius of* r.

Proof. Following the same procedure in obtaining the suboptimal solutions to general linear systems, we can easily prove Lemma 5 and we just omit it here. □

2.4 Robust MPC-based Consensus Strategy

In this section, a nonlinear state feedback (2.3) is designed via MPC to meet the aforementioned control requirements. We first present a nominal control framework, referred to as nominal MPC-based consensus control, by synthesizing the suboptimal linear consensus protocol obtained from Lemma 5. Next, by tightening the constraints into smaller sets to handle disturbance, a robust version of the MPC framework, referred to as robust MPC-based consensus control, is proposed.

2.4.1 Nominal MPC-based Consensus

Consider the nominal system

$$\boldsymbol{x}(k+1) = (I_M \otimes A)\boldsymbol{x}(k) + (I_M \otimes B)\boldsymbol{u}(k), \tag{2.47}$$

with the input constraints $\boldsymbol{u}(k) \in \mathcal{U}$. We want to find a nonlinear consensus protocol designed via MPC, such that the nominal system in (2.47) achieves consensus while satisfying the input constraints.

The nominal consensus control input is generated by solving the following optimization problem.

$$\begin{aligned}
\underset{\boldsymbol{c}(\cdot)}{\text{minimize}} \quad & J_N(\boldsymbol{x}(k), \boldsymbol{c}(k)) = \sum_{t=0}^{N-1} \|\boldsymbol{c}(k+t|k)\|^2 \\
\text{subject to:} \quad & \boldsymbol{x}(k|k) = \boldsymbol{x}(k), \\
& \boldsymbol{x}(k+t+1|k) = \Phi\boldsymbol{x}(k+t|k) + (I_M \otimes B)\boldsymbol{c}(k+t|k), \\
& \boldsymbol{u}(k+t|k) = (\mathcal{L} \otimes K)\boldsymbol{x}(k+t|k) + \boldsymbol{c}(k+t|k) \in \mathcal{U}, \\
& \boldsymbol{x}(k+N|k) \in \mathcal{X}^f.
\end{aligned}$$

The constant $N > 0$ is the prediction horizon, which represents the number of free state and control moves in the OCP. The optimal solution to the OCP is denoted by $\boldsymbol{c}^*(k) = \left[\boldsymbol{c}^*(k|k)^{\mathrm{T}}, \cdots, \boldsymbol{c}^*(k+N-1|k)^{\mathrm{T}}\right]^{\mathrm{T}}$, and the corresponding optimal predicted state and input sequences are given as follows:

$$\boldsymbol{x}^*(k) = \left[\boldsymbol{x}^*(k|k)^{\mathrm{T}}, \cdots, \boldsymbol{x}^*(k+N|k)^{\mathrm{T}}\right]^{\mathrm{T}}, \quad \boldsymbol{u}^*(k) = \left[\boldsymbol{u}^*(k|k)^{\mathrm{T}}, \cdots, \boldsymbol{u}^*(k+N-1|k)^{\mathrm{T}}\right]^{\mathrm{T}}.$$

As in many existing MPC works, only the first element of the optimal predicted control sequence

$$\boldsymbol{u}(k) = \boldsymbol{u}^*(k|k) = (\mathcal{L} \otimes K)\boldsymbol{x}(k|k) + \boldsymbol{c}^*(k|k),$$

is implemented to the plant. The predictive controller implicitly defines a nonlinear consensus input from the solution of the constrained convex optimization problem. To ensure recursive feasibility of the OCP, the terminal region $\mathcal{X}^f$ is determined offline, so that

1. The corresponding control variables vanish to zero and the input constraints

are satisfied, i.e.,

$$\boldsymbol{x} \in \mathcal{X}^f \Rightarrow \begin{cases} \boldsymbol{c} = 0, \\ \boldsymbol{u} = (\mathcal{L} \otimes K)\boldsymbol{x} \in \mathcal{U}; \end{cases}$$

2. The successive state is still inside the terminal region,

$$\boldsymbol{x} \in \mathcal{X}^f \Rightarrow (I_M \otimes A + \mathcal{L} \otimes BK)\boldsymbol{x} := \varPhi \boldsymbol{x} \in \mathcal{X}^f.$$

We define the feasible set for the terminal constraints in N-step horizon as follow:

$$\mathcal{F}_N = \{\boldsymbol{x} | (\mathcal{L} \otimes K)\varPhi^i \boldsymbol{x} \in \mathcal{U}, \forall i \in \mathbb{N}_{[0,N]}\}.$$

By taking $N \to \infty$, the terminal region becomes

$$\mathcal{X}^f = \lim_{N \to \infty} \mathcal{F}_N. \tag{2.48}$$

The terminal set (2.48) containing infinite number of constraints can be proved bounded [71]. For implementation issues, a finite integer n^* can be determined offline, such that $\mathcal{F}_\infty = \mathcal{F}_{n^*}$ [71].

2.4.2 Robust MPC-based Consensus

The constraints in the nominal MPC-based consensus problem involve only disturbance-free predictions. Therefore, we cannot guarantee that the real system state and input will satisfy the constraints due to the present of disturbance. Hence the nominal MPC-based consensus control is possible to lose feasibility and cannot provide any guarantee of stability. To this end, we propose a robust MPC-based consensus control framework to solve this, while preserve feasibility despite the existence of disturbance.

Considering the disturbed MAS in (2.2), the real state and control input are given by

$$\boldsymbol{x}(k+t) = \boldsymbol{x}(k+t|k) + \sum_{i=1}^{t} \Phi^{i-1}\boldsymbol{\omega}(k+t-1),$$
$$\boldsymbol{u}(k+t) = \boldsymbol{u}(k+t|k) + \sum_{i=1}^{t} (\mathcal{L}\otimes K)\Phi^{i-1}\boldsymbol{\omega}(k+t-1),$$

where the first terms on the right hand side of the equations represent the disturbance-free prediction and the latter terms represent the forced responses caused by disturbance.

Let

$$\mathcal{R}_j := \bigoplus_{i=0}^{j-1} \Phi^i \mathcal{W} \tag{2.49}$$

denote the j-step reachable set for the closed-loop system from the origin, driven by the bounded disturbance as the only input. Hence, a sufficient condition for the real input satisfying the input constraints, i.e., $\boldsymbol{u}(k+t) \in \mathcal{U}$, is to impose more stringent constraints on the nominal predictions, $\boldsymbol{u}(k+t|k) \in \mathcal{U}_t, t \geq 0$, where the restricted input constraint set is given by $\mathcal{U}_t := \mathcal{U} \ominus (\mathcal{L}\otimes K)\mathcal{R}_t$.

Problem 2.

$$\begin{aligned}
\underset{\boldsymbol{c}(\cdot)}{\text{minimize}} \quad & J_N(\boldsymbol{x}(k), \boldsymbol{c}(k)) = \sum_{t=0}^{N-1} \|\boldsymbol{c}(k+t|k)\|^2 \\
\text{subject to:} \quad & \boldsymbol{x}(k|k) = \boldsymbol{x}(k), \\
& \boldsymbol{x}(k+t+1|k) = \Phi\boldsymbol{x}(k+t|k) + (I_M \otimes B)\boldsymbol{c}(k+t|k), \\
& \boldsymbol{u}(k+t|k) = (\mathcal{L}\otimes K)\boldsymbol{x}(k+t|k) + \boldsymbol{c}(k+t|k) \in \mathcal{U}_t, \\
& \boldsymbol{x}(k+N|k) \in \mathcal{X}_N^f.
\end{aligned} \tag{2.50}$$

To ensure the satisfactory of the terminal constraint under perturbation, the restricted terminal region $\mathcal{X}_N^f := \mathcal{X}^f \ominus \mathcal{R}_N$ is applied.

In fact, if $\boldsymbol{u}(k+t|k) \in \mathcal{U} \ominus (\mathcal{L} \otimes K)\mathcal{R}_t$ holds, one immediately gets $\boldsymbol{u}(k+t|k) \oplus (\mathcal{L} \otimes K)\mathcal{R}_t \subset \mathcal{U}$ for all possible disturbance, which further implies the real control input satisfying the constraints, i.e., $\boldsymbol{u}(k+t) \in \boldsymbol{u}(k+t|k) \oplus (\mathcal{L} \otimes K)\mathcal{R}_t \subset \mathcal{U}$.

Based on the above arguments, the following robust MPC-based consensus framework is proposed.

The robust MPC-based consensus control algorithm is given below.

Algorithm 1 Robust MPC-based Consensus Algorithm

Require: initial state of the MAS $\boldsymbol{x}_0$; index $k = 0$; prediction horizon N.

1: **while** The control action is not stopped **do**

2: Measure the current states of all agents $\boldsymbol{x}(k)$ in the system (2.2);

3: Solve the optimal control problem in Problem 2, obtain the sequences $\boldsymbol{c}^*(k)$, $\boldsymbol{u}^*(k)$; the concatenated control input is taken as $\boldsymbol{u}(k) = \boldsymbol{u}^*(k|k)$;

4: Each agent applies the first element of the optimal control sequence

$$u_i(k) = u_i^*(k|k)$$

to the system in (2.1);

5: Increment $k = k + 1$, go back to step 2.

6: **end while**

2.5 Feasibility and Convergence Analysis

Concatenating the optimal "tail" at sampling time k with terminal zero elements, the control variable candidate sequence can be taken as

$$\begin{aligned}\tilde{\boldsymbol{c}}(k+1) &= \left[\tilde{\boldsymbol{c}}(k+1|k+1)^{\mathrm{T}}, \cdots, \tilde{\boldsymbol{c}}(k+N-1|k+1)^{\mathrm{T}}, \tilde{\boldsymbol{c}}(k+N|k+1)^{\mathrm{T}}\right]^{\mathrm{T}}, \\ &= \left[\boldsymbol{c}^*(k+1|k)^{\mathrm{T}}, \cdots, \boldsymbol{c}^*(k+N-1|k)^{\mathrm{T}}, 0\right]^{\mathrm{T}}.\end{aligned} \tag{2.51}$$

Definition 3. *A control sequence* $\boldsymbol{c}(k) = \left[\boldsymbol{c}(k|k)^{\mathrm{T}}, \cdots, \boldsymbol{c}(k+N-1|k)^{\mathrm{T}}\right]^{\mathrm{T}}$ *is said to* *be admissible for state $\boldsymbol{x}(k)$ if the constraints in (2.50) are satisfied. A state $\boldsymbol{x}(k)$ is said to be feasible if there exists at least one control variable sequence $\boldsymbol{c}(k)$ admissible for $\boldsymbol{x}(k)$.*

The following lemma and theorem provide sufficient conditions such that the robust MPC-based consensus control meets the control objectives.

Lemma 6. *For the MAS in (2.2) under the regulation of the robust MPC-based consensus control $\boldsymbol{u}(k) = (\mathcal{L} \otimes K)\boldsymbol{x}(k) + \boldsymbol{c}(k)$, the following implication holds*

$$\boldsymbol{c}^*(k) \text{ is admissible for } \boldsymbol{x}(k) \Rightarrow \tilde{\boldsymbol{c}}(k+1) \text{ is admissible for } \boldsymbol{x}(k+1).$$

Proof. The predictions corresponding to $\boldsymbol{c}^*(k)$ with $\boldsymbol{x}(k)$ are denoted by $\boldsymbol{u}^*(k)$ and $\boldsymbol{x}^*(k)$. We also denote the predictions for the control variable candidate $\tilde{\boldsymbol{c}}(k+1)$ with $\boldsymbol{x}(k+1)$ by $\tilde{\boldsymbol{u}}(k+1)$ and $\tilde{\boldsymbol{x}}(k+1)$. Since the optimal sequence $\boldsymbol{c}^*(k)$ is the solution to Problem 2 at time k, it follows that $\boldsymbol{u}^*(k+t|k) \in \mathcal{U}_t$ and $\boldsymbol{x}^*(k+N|k) \in \mathcal{X}_N^f$. Due to the existence of disturbance, we can check that

$$\begin{aligned}
\tilde{\boldsymbol{x}}(k+1+t|k+1) &= \boldsymbol{x}^*(k+1+t|k) + \Phi^t \boldsymbol{\omega}(k) \in \mathcal{X}_{t+1}^f \oplus \Phi^t \mathcal{W}, \\
\tilde{\boldsymbol{u}}(k+1+t|k+1) &= \boldsymbol{u}^*(k+1+t|k) + (\mathcal{L} \otimes K)\Phi^t \boldsymbol{\omega}(k) \in \mathcal{U}_{t+1} \oplus (\mathcal{L} \otimes K)\Phi^k \mathcal{W}.
\end{aligned}$$

By the properties of the Minkowski sum, it holds that

$$\begin{aligned}
\tilde{\boldsymbol{x}}(k+1+N|k+1) \in \mathcal{X}_{N+1}^f \oplus \Phi^N \mathcal{W} &= (\mathcal{X}_N^f \ominus \mathcal{R}_{N+1}) \oplus \Phi^N \mathcal{W} \\
&= (\mathcal{X}_N^f \ominus (\mathcal{R}_N \oplus \Phi^N \mathcal{W})) \oplus \Phi^N \mathcal{W} \\
&\subseteq \mathcal{X}^f \ominus \mathcal{R}_N = \mathcal{X}_N^f.
\end{aligned}$$

Similarly, we can prove that the control input candidate satisfies the input constraints:

$$\tilde{\boldsymbol{u}}(k+1+t|k+1) \in \mathcal{U}_{t+1} \oplus (\mathcal{L} \otimes K)\Phi^k \mathcal{W} \subseteq \mathcal{U}_t.$$

Combining the above constraint satisfactory results, we can conclude that Lemma 6 holds. □

Theorem 2. *Given that the initial state $\boldsymbol{x}_0$ is feasible, the MAS in (2.2) under the control of the proposed robust MPC-based consensus protocol, with $\boldsymbol{c}(k) = \boldsymbol{c}^*(k|k)$, satisfies the following properties:*

1. $\lim_{k\to\infty} \boldsymbol{c}(k) = 0$*;*
2. *consensus is not guaranteed due to disturbances, but the disagreement among the agents can be reduced to a bounded level.*

Proof. Denote the Lyapunov function by

$$V(k) := J^*(\boldsymbol{c}(k)) = \sum_{t=0}^{N-1} \|\boldsymbol{c}^*(k+t|k)\|^2 \geq 0,$$

and we evaluate its value for the candidate $\tilde{\boldsymbol{c}}(k+1)$:

$$\tilde{V}(k+1) = \sum_{i=0}^{N-1} \|\tilde{\boldsymbol{c}}(k+1+i|k+1)\|^2 = -\|\boldsymbol{c}^*(k|k)\|^2 + V(k).$$

The candidate $\tilde{\boldsymbol{c}}(k+1)$ is suboptimal to Problem 2 at time instant $k+1$, so it holds that

$$\begin{aligned} &\tilde{V}(k+1) = V(k) - \|\boldsymbol{c}^*(k|k)\|^2 \leq V(k+1), \\ \Rightarrow &V(k+1) - V(k) \leq -\|\boldsymbol{c}^*(k|k)\|^2 \leq 0. \end{aligned}$$

Considering the sum of the above inequalities, one gets

$$\lim_{k\to\infty}\sum_{t=0}^{k}(V(k+1)-V(k)) = \lim_{k\to\infty} V(k+1)-V(0) = \lim_{k\to\infty}\sum_{t=0}^{k} - \left\|\boldsymbol{c}^*(t|t)\right\|^2.$$

Since $0 \le V(k+1) \le \infty$ and $0 \le V(0) \le \infty$, we obtain that

$$0 \le \lim_{k\to\infty} V(k+1) = V(0) - \lim_{k\to\infty}\sum_{t=0}^{k}\left\|\boldsymbol{c}^*(t|t)\right\|^2 \le \infty,$$

which implies $\lim\limits_{k\to\infty}\sum_{t=0}^{k}\left\|\boldsymbol{c}^*(t|t)\right\|^2 \le \infty$, and further we obtain $\lim\limits_{k\to\infty}\left\|\boldsymbol{c}^*(k|k)\right\|^2 = 0$, which proves the first part of Theorem 2. Accordingly, we can summarize the convergence results:

$$\begin{aligned}\lim_{k\to\infty}\boldsymbol{x}(k) = \lim_{k\to\infty}\{&\Phi^k\boldsymbol{x}_0 + \sum_{t=1}^{k}\Phi^{t-1}(I_M\otimes B)\boldsymbol{c}^*(k-t|k-t)\\ &+\sum_{t=1}^{k}\Phi^{t-1}\boldsymbol{\omega}(k-t)\} \subseteq \mathcal{R}_k \subset \mathcal{R}_\infty,\end{aligned}$$

where the proof for the set $\mathcal{R}_\infty$ being bounded is given in the previous discussion. This completes the second part of the proof. □

2.6 Numerical Examples

In this section we use several numerical examples to illustrate the proposed MPC-based consensus control framework. Consider a group of four identical oscillators given by the following linear dynamics:

$$x_i(k+1) = Ax_i(k) + Bu_i(k) + \omega_i(k), \quad x_i(0) = x_{i0}, \quad i \in \mathbb{N}_{[1,4]},$$

where $A=[0,1;-1,0]$ and $B=[0.5;0.5]$. The initial states of the 4 agents are chosen as

$$x_{10}=\begin{bmatrix}-0.08\\ 0.11\end{bmatrix}, x_{20}=\begin{bmatrix}0.12\\ -0.08\end{bmatrix}, x_{30}=\begin{bmatrix}-0.09\\ -0.14\end{bmatrix}, x_{40}=\begin{bmatrix}-0.12\\ 0.04\end{bmatrix}.$$

Assume the graph underlying the MAS is undirected and the Laplacian matrix is $\mathcal{L}=[1,-1,0,0;-1,2,-1,0;0,-1,2,-1;0,0,-1,1]$. The smallest nonzero and largest eigenvalue of $\mathcal{L}$ are $\lambda_2=0.5858$ and $\lambda_4=3.4142$. We make use of the cost functional,

$$J(x,u)=\sum_{k=0}^{\infty}x(k)^{\mathrm{T}}(\mathcal{L}\otimes Q)x(k)+u(k)^{\mathrm{T}}(I_4\otimes R)u(k),$$

to evaluate the level of consensus. Given the constant ϵ, we compute a positive definite solution P by solving the modified DARE

$$A^{\mathrm{T}}PA-(c^2\lambda_4^2-2c\lambda_4)A^{\mathrm{T}}PB(R+B^{\mathrm{T}}PB)^{-1}B^{\mathrm{T}}PA+\lambda_4 Q-P+\epsilon I_4=0,$$

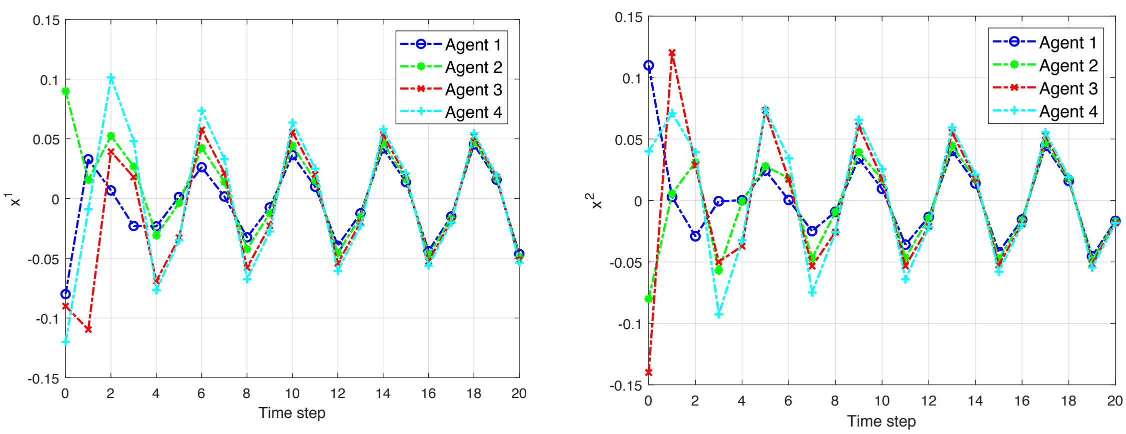
where $c=2/(\lambda_2+\lambda_4)$. The other parameters adopted in this numerical example are listed in Table 2.1.

Table 2.1: The parameters adopted in our work.

Prediction horizon N	10
Weighting matrix Q	[2,0;0,1]
Weighting matrix R	1
Scalar in the modified DARE ϵ	0.001
Finite integer for the terminal region n^*	5
Solution to the DARE P	[9.1328,1.1642;1.1642,9.1328]
Feedback gain K	[-0.4052;0.2078]
The disturbance bound $\|\omega_i\|_\infty$	≤ 0.003
Input constraints $\|u_i\|_\infty$	≤ 0.1

In the following example for the oscillator MAS, we first show the performance of the proposed linear consensus protocol developed by the suboptimal linear quadratic approach. Then we demonstrate the effectiveness of our tube-based MPC consensus control in handling input constraints when bounded disturbance is involved.

The plots of the four oscillators with the suboptimal linear feedback are shown in Figure 2.1. The states of the agents gradually agree on a common online trajectory, but the size of some control inputs exceed the restriction of 0.1 (see Figure 2.2).



(a) State trajectories for x^1 coordinates (b) State trajectories for x^2 coordinates

Figure 2.1: State trajectories for the oscillators regulated by the linear consensus protocol

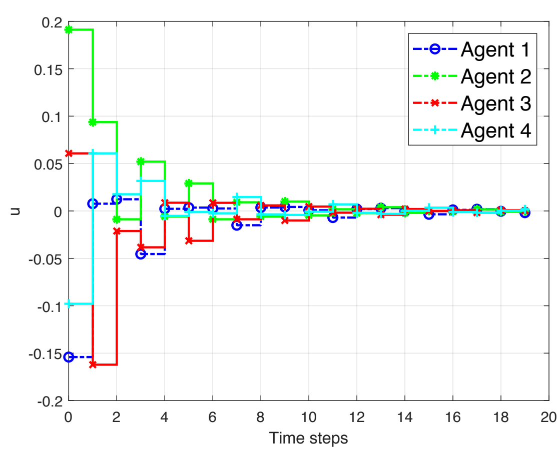
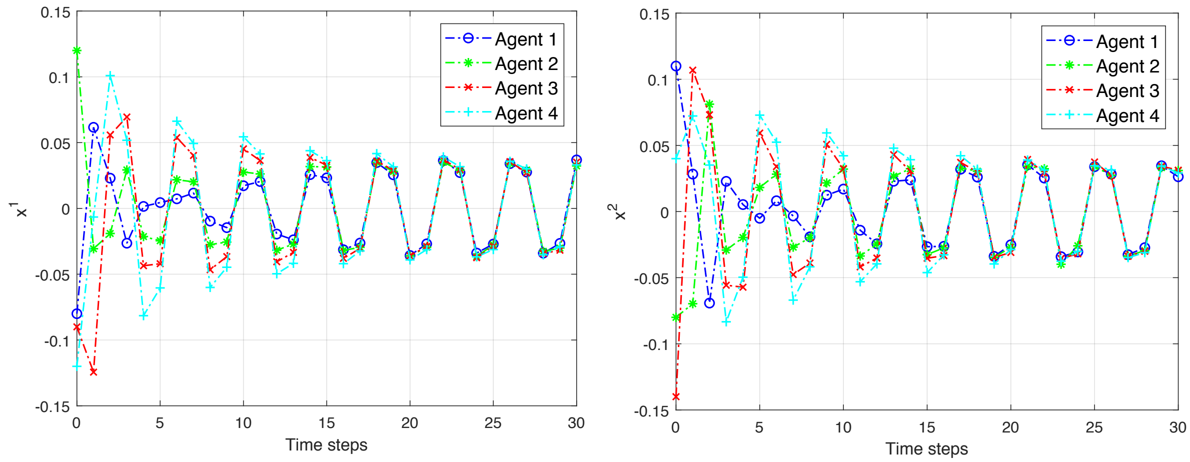


Figure 2.2: Control inputs of the linear consensus protocol



(a) State trajectories for x^1 coordinates (b) State trajectories for x^2 coordinates

Figure 2.3: State trajectories for the oscillators regulated by the MPC inputs

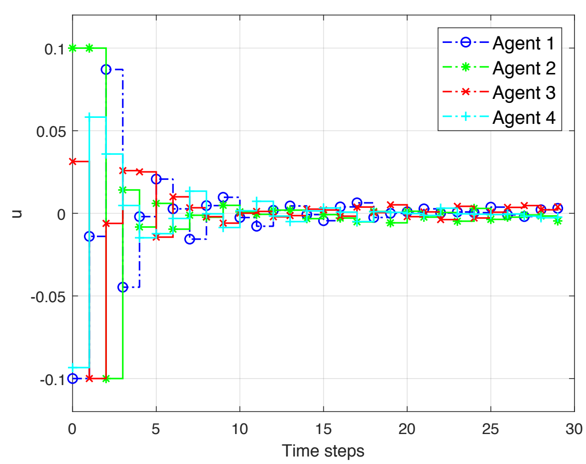


Figure 2.4: MPC inputs for all agents

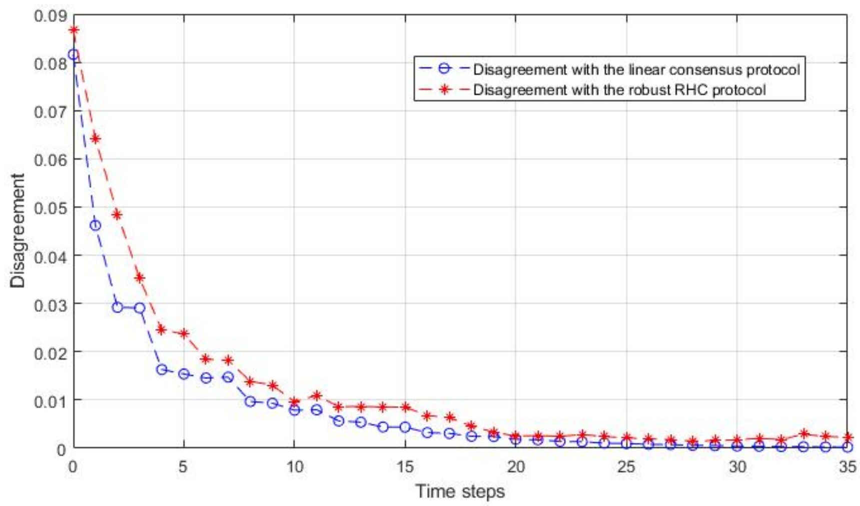
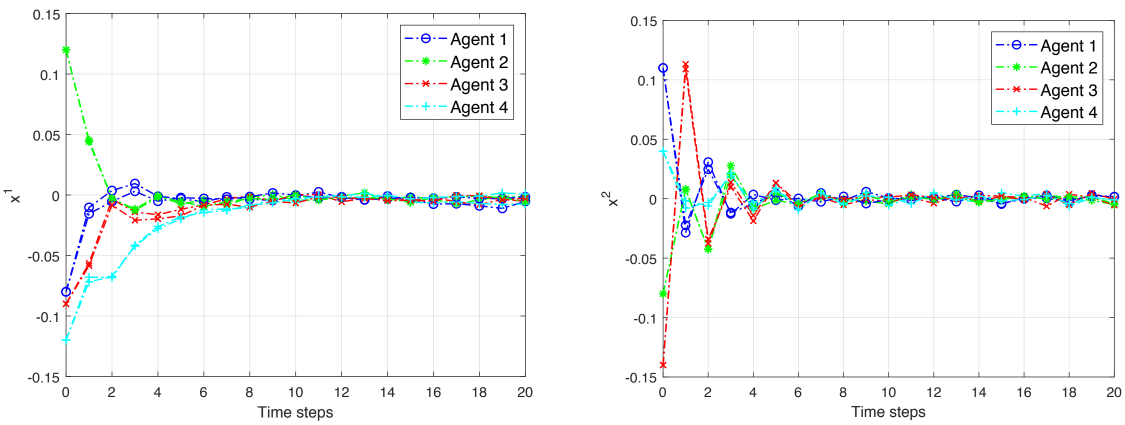


Figure 2.5: Disagreement among the agents

Figure 2.3 shows that the disagreement among the agents is gradually reduced, but cannot diminish to zero due to the existence of disturbance (see the red dashed line in Figure 2.5). This is also true for the MPC inputs. Thanks to the stringent constraints in prediction, the robust MPC-based consensus protocol is recursively feasible at each time instant, and the control inputs implemented to all agents satisfy the input saturation (as shown in Figure 2.4). Figure 2.5 compares the disagreement convergence rate for the linear and robust MPC-based consensus protocol. It is well noted that the robust MPC-based consensus algorithm converges slower than the linear counterpart, for its sacrifice of preserving input constraint satisfaction.

It is worthwhile to mention that this MPC framework is also applicable to stable and unstable MASs. Consider an MAS of the same form of dynamics as mentioned at the beginning of this section. We further choose the system parameters of the stable and unstable dynamics as $A_1 = [0.8, 0; 0, -0.7]$, $B_1 = [0.5; 0.5]$; $A_2 = [0, 1; -1.1, 0]$, $B_2 = [0.5; 0.5]$, respectively. Following the same line, the feedback gain matrices for these two cases can be obtained: $K_1 = [-0.4052, 0.2078]$, $K_2 = [0.3974, -0.4958]$.



(a) State trajectories for x^1 coordinates of the stable MAS

(b) State trajectories for x^2 coordinates of the stable MAS

Figure 2.6: State trajectories for the stable MAS regulated by the MPC inputs

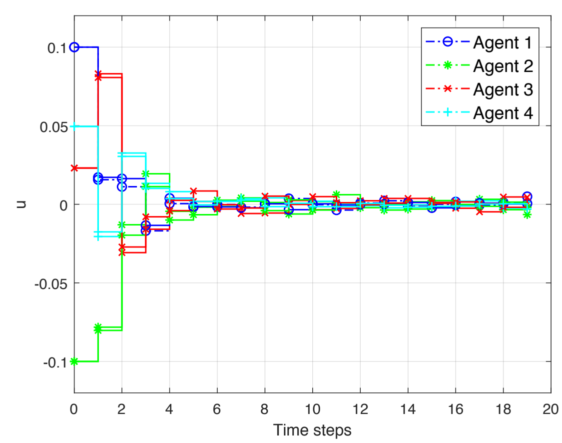
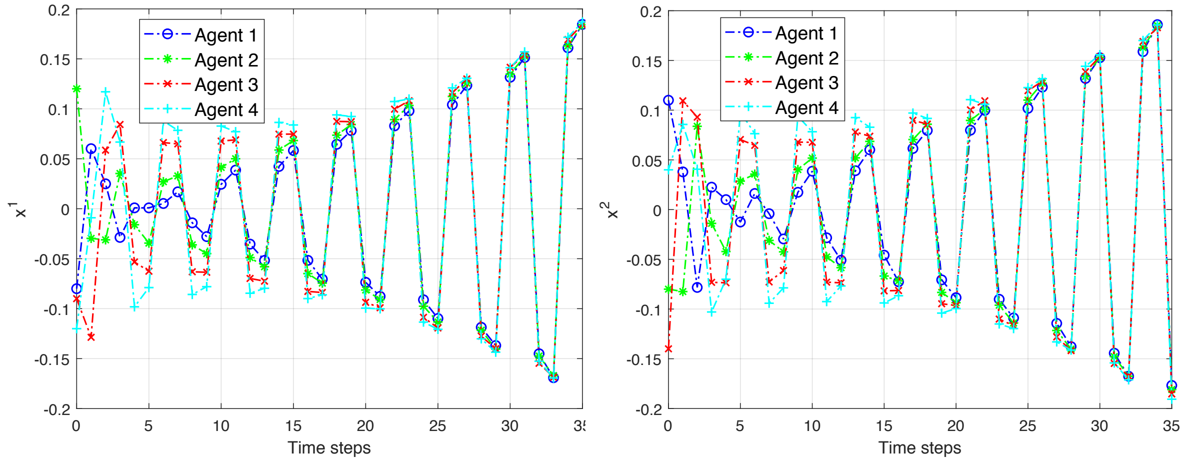


Figure 2.7: MPC inputs of the stable MAS



(a) State trajectories for x^1 coordinates of the unstable MAS

(b) State trajectories for x^2 coordinates of the unstable MAS

Figure 2.8: State trajectories for the unstable MAS regulated by the MPC inputs

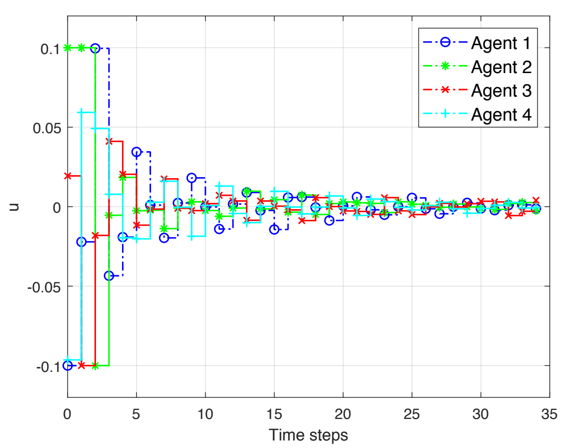


Figure 2.9: MPC inputs of the unstable MAS

We apply the proposed robust MPC framework to the above two types of systems with the same setup as the aforementioned oscillator case. The simulation results for stable and unstable MASs are presented in Figure 2.6-2.7 and Figure 2.8-2.9, respectively. All the results demonstrate that our consensus control strategy can reduce the disagreement among agents for both stable and unstable MASs. In the meanwhile, the input constraints are satisfied in both cases. Due to the persistent existence of disturbance, the disagreement, as well as the control inputs, cannot vanish to zeros, but can be limited to a bounded level.

2.7 Conclusion

In this chapter, a robust MPC-based consensus control scheme is proposed. The agent dynamics discussed in this chapter are governed by general discrete-time linear dynamics with bounded additive disturbance. A suboptimal linear consensus protocol is first introduced, followed by the associated nominal MPC strategy for expository of the offline computed constraint sets. Later we adopt a tube-based MPC consensus framework by tightening the original constraint sets, such that the actual system still satisfies the constraints under disturbance. Simulation results for three types of MASs (oscillator, stable an unstable) are provided to demonstrate the effectiveness of our control strategy.

Chapter 3

Distributed Model Predictive Control based Consensus of General Linear Multi-agent Systems with Input constraints

3.1 Introduction

In the study of MASs, cooperative control has been one of the most significant topics for the past decades, due to its wide applications in completing complex tasks by multiple robots, controllers and terminal devices in large scale networks. Especially, consensus is one of the most basic problems in multi-agent cooperation study. Many decent results have been proposed in recent years, for example in [72, 28, 73], providing a wide range of consensus protocols in different forms for multiple types of systems and communication networks.

Model predictive control (MPC), or namely, receding horizon control (RHC) techniques have received increasing attention due to their remarkable advantages in handling multi-variables and hard constraints. Distributed MPC algorithms have been adopted in MAS and some remarkable results can be found in [53, 58, 74, 75]. Among the MPC-based consensus algorithms, the majority of the existing works mainly focus on the stabilization of an a priori known set point [32, 52, 39] or trajectory tracking [5]. A decent work on disturbed nonlinear MAS stabilization via distributed MPC can be found in [55], where a novel constraint providing both robustness and convergence is proposed. However, consensus problem is more challenging as it requires all agents in an MAS agreeing a common online trajectory, in contrast to following an a priori given reference trajectory as formation or tracking control. Until now, only very few exceptions consider other cooperative control tasks like consensus.

Starting from simple systems, the researchers in [22, 23] implement the MPC schemes to solve consensus problem for unconstrained single and double-integrator dynamics. A novel distributed MPC controller for double-integrator systems is reported in [23], where the consensus problem is treated as static formation to an autonomous leader agent. In [58], consensus for general linear MASs is addressed under necessary and sufficient conditions by involving an unconstrained convex optimization framework. The algorithms developed in [60] and [76] solve nonlinear MAS consensus with state and input constraints, but iterative communication of both state and input information among agents and repeated optimization problem solving are required. It is worth noting that [27, 38] consider consensus for linear systems with constraints. However, a rather strong assumption on information exchange is made, where each agent is able to access its neighbors states simultaneously. This makes the MPC-based consensus algorithms not fully distributed and is especially troublesome for the MASs equipped with wireless networked digital controllers.

Therefore, the motivation of this chapter is to design a novel distributed consensus protocol via MPC for linear MASs with input constraints, avoiding iterative communication in transmission networks, as well as being truly distributed and applicable to networked digital controllers.

The main contributions of this chapter are mainly twofold.

- A novel distributed MPC-based consensus protocol and the conditions for designing such consensus algorithm for semi-stable MASs are proposed. At every time step, each agent only measures its current state, and makes use of the assumed neighbor state information received at previous instant, rather than accessing the current state of neighbors, then solves a local constrained convex optimization problem to generate the control input. The local optimization problem is solved and state information is broadcast among neighbors only once at each time instant.

- A pre-stabilizing linear consensus protocol is first introduced, such that the closed-loop system is asymptotically stable with respect to a consensus set. By minimizing the gap between the MPC input and the pre-stabilizing linear consensus control input, cooperation among agents is reinforced with satisfaction of the input constraints. With the properly designed bounds on the difference between the actual state and the assumed state, the overall system converges to a neighboring set of the consensus set.

Notation: Denote the field of real numbers by $\mathbb{R}$ and the field of non-negative real numbers by $\mathbb{R}_{\geq 0}$. We also denote any n dimensional column vector by $x \in \mathbb{R}^n$ and any $n \times n$ matrix by $A \in \mathbb{R}^{n\times n}$. Let $A \otimes B \in \mathbb{R}^{np\times mq}$ be the Kronecker product of two matrices where $A \in \mathbb{R}^{n\times m}$ and $B \in \mathbb{R}^{p\times q}$. For any scalar $a \in \mathbb{R}$, $|a|$ is the absolute value of a. For any vector $x_i \in \mathbb{R}^n$, $\|x_i\|$ and $\|x_i\|_\infty$ are the 2-norm and the infinity-norm of x_i, respectively. Let I_n be the $n \times n$ identity matrix. For a set

$\mathcal{X}_i \subseteq \mathbb{R}^n$ and a point $x_i \in \mathbb{R}^n$, the distance from the point to the set is defined as $|x_i|_{\mathcal{X}_i} := \inf_{z \in \mathcal{X}_i} \|x_i - z\|$. A function $\alpha(t) : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a class $\mathcal{K}$ function if α is continuous, monotonically increasing and $\alpha(0) = 0$. And if $\alpha(t) \to \infty$ as $t \to \infty$, it is of class $\mathcal{K}_\infty$. A function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a $\mathcal{KL}$ function if $\beta(\cdot, t)$ is a $\mathcal{K}$ function for every given t, and $\beta(\gamma, t)$ is monotonically decreasing to 0 as $t \to \infty$ for every given $\gamma \geq 0$. A function $\sigma : \mathbb{R} \to \mathbb{R}_{\geq 0}$ belongs to class $\mathcal{PD}$ (positive definite), if it is continuous, $\sigma(0) = 0$, and always positive otherwise. $\mathbf{1}_M$ represents an unit column vector of proper dimension whose M entries are all one(s). spec(A) is the spectrum of matrix A. A sequence of integers is denoted by $\mathbb{N}_{[m,n]} = \{m, m+1, \ldots, n\}$. The Cartesian product of multiple sets is denoted by $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_M$. We denote the largest and the smallest eigenvalue of a matrix $P \in \mathbb{R}^{n \times n}$ by $\lambda_{\max}(P)$ and $\lambda_{\min}(P)$, respectively. Given two sets A, $B \subseteq \mathbb{R}^n$ and vectors $a, b, c \in \mathbb{R}^n$, the Minkowski sum of the sets is defined by $A \oplus B := \{c | c = a + b, a \in A, b \in B\}$ and the Pontryagin difference of the two sets is denoted by $A \ominus B := \{c | c + b \in A, b \in B\}$. The Minkowski sum of multiple sets is given by $A_1 \oplus A_2 \cdots \oplus A_M = \bigoplus_{i=1}^M A_i$. The Drazin inverse of a matrix A is denoted by $A^{\#}$.

3.2 Preliminaries and Problem Statement

Consider an MAS consisting of M identical agents given by

$$x_i(k+1) = Ax_i(k) + Bu_i(k), \quad x_i(0) = x_{i0}, \quad i \in \mathbb{N}_{[1,M]}, \tag{3.1}$$

where $x_i(k) \in \mathbb{R}^n$ and $u_i(k) \in \mathcal{U}_i \subset \mathbb{R}^m$ are the system state and control input at time k, respectively. The compact set $\mathcal{U}_i$, containing the origin as its interior point, stands for the input constraint set. A system is said to be semi-stable if spec$(A) \leq 1$, and if spec$(A) = 1$, then 1 is a simple eigenvalue of A. In this chapter, we consider a linear semi-stable MAS given in (3.1).

3.2.1 Basic Concepts from Graph Theory

In the study of MASs, graph theory is often used to model information exchange among agents.

The communication topology underlying the MAS in (3.1) is characterized by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, with $\mathcal{V} = \{v_1, v_2, \cdots, v_M\}$ being the non-empty vertex set of M nodes and $\mathcal{E} = \{(v_i, v_j)\,|v_i, v_j \in \mathcal{V}, i \neq j\} \subset \mathcal{V} \times \mathcal{V}$ being the edge set. An edge $(v_i, v_j) \in \mathcal{E}$ represents a communication channel from agent i to j. In the corresponding adjacency matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{M\times M}$, $a_{ij} = 1$ stands for the existence of a communication pathway from agent i to j for all edges in $\mathcal{E}$; if $(v_i, v_j) \notin \mathcal{E}$ or otherwise, then $a_{ij} = 0$. An agent j is said to be a neighbor of agent i if the edge $(v_i, v_j) \in \mathcal{E}$, or namely $a_{ij} = 1$. Let $\mathcal{N}_i$ be the set of all neighboring vertices of node i, i.e., $\mathcal{N}_i := \{v_j|v_i, v_j \in \mathcal{V}, (v_i, v_j) \in \mathcal{E}, i \neq j\} \subseteq \mathcal{V}$ and $d_i := |\mathcal{N}_i|$ its cardinality.

In this chapter, we assume that the graph $\mathcal{G}$ is undirected, i.e., $a_{ij} = 1 \Leftrightarrow a_{ji} = 1$ and each agent has at least one neighbor. The Laplacian matrix of the graph $\mathcal{G}$ is denoted by $\mathcal{L} = \mathcal{D} - \mathcal{A}$, where the diagonal matrix $\mathcal{D} := \mathrm{diag}\{d_{ii}\} \in \mathbb{R}^{M\times M}$, with $d_{ii} = d_i$, $\forall i \in \mathbb{N}_{[1,M]}$, is the degree matrix containing the information about the number of edges attached to each vertex.

Definition 4. *A path from vertex i_1 to i_k is denoted by an edge sequence*

$$\{(i_1, i_2), (i_2, i_3) \cdots (i_{k-1}, i_k)\},$$

with all edges in the sequence $(i_{j-1}, i_j) \in \mathcal{E}$, $\forall j \in \mathbb{N}_{[1,k]}$. If there exists a vertex i such that any other vertices in graph $\mathcal{G}$ can be reached via at least one path, the graph $\mathcal{G}$ is said to contain a spanning tree and to be connected.

Given an undirected graph $\mathcal{G}$, its Laplacian matrix $\mathcal{L}$ is symmetric and simple, and consequently has real eigenvalues. Furthermore, the eigenvalues of $\mathcal{L}$ are always non-negative and can be reorganized in increasing order as $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{M-1} \leq \lambda_M$. Specially, for a connected graph $\mathcal{G}$, it always holds that $\lambda_1 = 0$. Since the Laplacian matrix $\mathcal{L}$ has zero row sums, it follows that $\mathbf{1}_M$ is an eigenvector with $\lambda_1 = 0$ as the associated eigenvalue, which indicates that $\mathcal{L}\mathbf{1}_M = \mathbf{0}$.

3.2.2 Preliminaries on Multi-agent Consensus

Consider the MAS in (3.1) underlying an undirected graph $\mathcal{G}$, the definition of consensus is given as follow.

Definition 5. *For all initial states x_i and x_j, the MAS over an undirected communication graph $\mathcal{G}$, is said to reach consensus if*

$$\|x_i(k) - x_j(k)\| \to 0 \text{ as } k \to \infty,$$

where $\forall i,\ j \in \mathbb{N}_{[1,M]},\ i \neq j$.

According to Definition 5, the MAS achieving consensus means that the states of all agents are identical, including the scenarios where the agents are stabilized simultaneously to an a priori known static point.

The following assumptions are made in the reminder of this chapter, such that the necessary conditions for multi-agent consensus are satisfied [17].

Assumption 2. *The identical pair $(A, B) \in \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times m}$ in the MAS given by (3.1) is assumed to be controllable.*

Assumption 3. *The graph $\mathcal{G}$ associated with the MAS system in (3.1) is connected.*

3.2.3 Set-wise Stabilization

For convenience, we can write (3.1) in compact form as

$$\boldsymbol{x}(k+1) = (I_M \otimes A)\boldsymbol{x}(k) + (I_M \otimes B)\boldsymbol{u}(k), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{3.2}$$

where the compact state and control input are denoted by

$$\boldsymbol{x}(k) := [x_1^{\mathrm{T}}(k), x_2^{\mathrm{T}}(k), \cdots, x_M^{\mathrm{T}}(k)]^{\mathrm{T}} \in \mathbb{R}^{n\times M},$$

$$\boldsymbol{u}(k) := [u_1^{\mathrm{T}}(k), u_2^{\mathrm{T}}(k), \cdots, u_M^{\mathrm{T}}(k)]^{\mathrm{T}} \in \mathcal{U} \subseteq \mathbb{R}^{m\times M},$$

respectively. The input constraint set of the overall system is defined by the Cartesian product of multiple single input constraint sets as $\mathcal{U} := \mathcal{U}_1 \times \cdots \times \mathcal{U}_M$.

Viewed from the perspective of set theory, multi-agent consensus can be characterized as a set-wise stabilization problem. To this end, a consensus set for the MAS in (3.2) is introduced:

$$\mathcal{C} = \{\boldsymbol{x} | x_1 = x_2 = \cdots = x_M\},$$

where $\boldsymbol{x} \in \mathbb{R}^{n\times M}$ is the compact state vector of the overall system.

When the MAS in (3.2) achieves consensus, the state $\boldsymbol{x}(k)$ reaches the consensus set $\mathcal{C}$. It follows that the distance between the state and the consensus set becomes zero, i.e., $|\boldsymbol{x}(k)|_{\mathcal{C}} = \inf_{\boldsymbol{z}\in\mathcal{C}} \|\boldsymbol{x}(k) - \boldsymbol{z}\| = 0$. Therefore, the necessity to study set-wise Lyapunov stability naturally arises. We are now in a position to introduce some definitions and preliminaries for set stability.

Definition 6 ([38])**.** *For a discrete-time autonomous system*

$$x^+ = f(x), \tag{3.3}$$

where $x \in \mathbb{R}^n$ *and* $f : \mathbb{R}^n \to \mathbb{R}^n$ *denotes a continuous function. The solution to (3.3) at time* k *with the initial state* x_0 *is denoted by* $\varphi(k; x_0)$*. A non-empty closed subset of* $\mathbb{R}^n$*, denoted by* $\mathcal{O}$*, is forward invariant if, for any given initial state* $x_0 \in \mathcal{O}$*, it holds that*

$$\varphi(k; x_0) \in \mathcal{O}, \quad \forall k \geq 0.$$

Note that the forward invariant set $\mathcal{O}$ is not necessarily bounded. Similar to asymptotic stability with respect to a static point, we recall the definition of set-wise asymptotic stability given in [38] and a lemma in [77].

Definition 7 ([38])**.** *The system in (3.3) is asymptotically stable with respect to a forward invariant set* $\mathcal{O}$ *if the following conditions are satisfied:*

- *(set-wise Lyapunov stability) for every* $\epsilon > 0$*, a scalar* $\delta > 0$ *can always be found, such that* $|x_0|_{\mathcal{O}} < \delta$ *implies* $|\varphi(k; x_0)| < \epsilon$*,* $\forall k \geq 0$*;*

- *(set-wise attraction) for* $x_0 \in \mathcal{X}_i \subseteq \mathbb{R}^n$*,* $|\varphi(k; x_0)|_{\mathcal{O}} \to 0$ *as* $k \to \infty$*.*

Lemma 7 ([77])**.** *If there exists a Lyapunov function* $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ *for the system in (3.3) and the given forward invariant set* $\mathcal{O}$*, such that*

- $\alpha_1(|x|_{\mathcal{O}}) \leq V(x) \leq \alpha_2(|x|_{\mathcal{O}})$*,*

- $V(x^+) - V(x) \leq -\alpha_3(|x|_{\mathcal{O}})$*,*

for any $x_0 \in \mathcal{X}_i \subseteq \mathbb{R}^n$*, where functions* α_1*,* α_2 *belong to class* $\mathcal{K}$ *and* $\alpha_3 \in \mathcal{PD}$*, then we can conclude that the system in (3.3) is asymptotically stable with respect to the forward invariant set* $\mathcal{O}$*.*

3.2.4 Problem Statement

Consider a linear semi-stable MAS given by (3.2), the distributed controller

$$u_i^l(k) = K \sum_{j \in \mathcal{N}_i} (x_i(k) - x_j(k)), \tag{3.4}$$

proposed in [38], can drive the system state to the consensus set $\mathcal{C}$, when there is no constraint involved. However, implementing (3.4) to a digital controller is difficult, since it requires every agent to have continuous access to their neighbors states $x_j(k)$ simultaneously at every discrete time instant. To relax these requirements, we are more interested in designing distributed MPC-based consensus strategy,

$$u_i(k) = g_i(x_i(k), \hat{x}_j(k)), \tag{3.5}$$

where the continuous nonlinear function $g_i : \mathbb{R}^n \to \mathbb{R}^m$, such that the closed-loop system can be driven to the consensus set, while satisfying input constraints. The assumed state of agent j at time k is denoted by $\hat{x}_j(k)$ and will be specified later. For convenience, we introduce a control variable

$$c_i(k) := u_i(k) - u_i^l(k) = u_i(k) - K \sum_{j \in \mathcal{N}_i} (x_i(k) - \hat{x}_j(k)) \tag{3.6}$$

to characterize the difference between the distributed MPC input (3.5) and the pre-stabilizing linear consensus control input (3.4). Accordingly the closed-loop systems become

$$x_i(k+1) = Ax_i(k) + BK \sum_{j \in \mathcal{N}_i} (x_i(k) - \hat{x}_j(k)) + Bc_i(k), \quad \forall i, j \in \mathbb{N}_{[1,M]}, i \neq j. \tag{3.7}$$

By minimizing the control variable, the MPC input gradually approaches the pre-stabilizing linear consensus control input, indicating that the cooperation among agents is reinforced. It is worth noting that, the assumed neighbor state $\hat{x}_j(k)$ is involved in the distributed MPC controller design, as it is not realistic for digital decision makers in a wireless digital networked system to grant continuous access to state of neighbors simultaneously. To this end, restricting the deviation between the real and the assumed states becomes necessary. The assumed state should lie in a specified neighborhood of the real state,

$$\hat{x}_i(k) \in x_i(k) \oplus \boldsymbol{\varepsilon}_i, \tag{3.8}$$

where $0 \in \boldsymbol{\varepsilon}_i \subset \mathbb{R}^n$, so that consistency can be ensured between the intended behavior of an agent and what its neighbors believe how the agent will behave. We are now in the position to present our distributed MPC-based consensus algorithm.

3.3 Distributed MPC-based Consensus

Given the prediction horizon N, the cost function for agent i at time k can be formulated as follow:

$$J_N^i(x_i(k), \hat{\boldsymbol{x}}_{\boldsymbol{j}}(k), c_i(k)) = \sum_{t=0}^{N-1} \|c_i(k+t|k)\|^2, \tag{3.9}$$

where $\hat{\boldsymbol{x}}_{\boldsymbol{j}}(k) = \begin{bmatrix} \hat{x}_{j_1}^{\mathrm{T}}(k) & \hat{x}_{j_2}^{\mathrm{T}}(k) & \cdots & \hat{x}_{j_{d_i}}^{\mathrm{T}}(k) \end{bmatrix}^{\mathrm{T}}$ is the collection of assumed state sequences of the neighboring agents and $d_i = |\mathcal{N}_i|$. The distributed optimal control problem $\mathcal{P}_i$ is given by:

$$\underset{c_i(\cdot)}{\text{minimize}} \quad J_N^i(x_i(k), \hat{\boldsymbol{x}}_{\boldsymbol{j}}(k), c_i(k)) \tag{3.10a}$$

$$\text{subject to} \quad x_i(k|k) = x_i(k), \tag{3.10b}$$

$$u_i(k+t|k) = K \sum_{j \in \mathcal{N}_i} (x_i(k+t|k) - \hat{x}_j(k+t|k)) + c_i(k+t|k), \tag{3.10c}$$

$$u_i(k+t|k) \in \mathcal{U}_i, \tag{3.10d}$$

$$x_i(k+t+1|k) = Ax_i(k+t|k) + Bu_i(k+t|k), \tag{3.10e}$$

$$\|x_i(k+t|k) - \hat{x}_i(k+t|k)\| \leq \Delta_i, \tag{3.10f}$$

$$x_i(k+N|k) \in \mathcal{X}_i^f, \tag{3.10g}$$

where $t \in \mathbb{N}_{[0,N-1]}$ and $\Delta_i > 0$ is a positive constant. The terminal region defined as

$$\mathcal{X}_i^f = \{x_i \in \mathbb{R}^n | x_i^{\mathrm{T}} \sum_{j \in \mathcal{N}_i} S_2(x_i - \hat{x}_j(k+N|k)) \leq \frac{\beta_s^2}{M}\},$$

is a forward invariant set [38]. The gain matrix K, the positive semi-definite weighting matrix $S_2 \in \mathbb{R}^{n \times n}$ and the positive constant β_s are given in [38]. The solution to $\mathcal{P}_i$ is denoted by $c_i^*(k+t|k) = \arg\min J_N^i(x_i(k), \hat{\boldsymbol{x}}_{\boldsymbol{j}}(k), c_i(k)), t \in \mathbb{N}_{[0,N-1]}$ and the corresponding optimal predicted state and input are denoted by $x_i^*(k+t|k), t \in \mathbb{N}_{[0,N]}$ and $u_i^*(k+t|k), t \in \mathbb{N}_{[0,N-1]}$, respectively. At time instant k, we make use of the predicted state sequence received at previous time step to construct the assumed state trajectory as

$$\hat{x}_j(k+t|k) = \begin{cases} x_j^*(k+t|k-1), & t \in \mathbb{N}_{[0,N-1]}, \\ Ax_j^*(k+N-1|k-1), & t = N. \end{cases} \tag{3.11}$$

This construction can be done once the local $\mathcal{P}_j$ is solved and the sequence $x_j^*(k-1+t|k-1)$, $t \in \mathbb{N}_{[0,N]}$ is available.

Remark 4. *The constraint (3.10f) bounds the optimal predicted state sequence $x_i^*(k+t|k)$ within a tube centered in the assumed state sequence $\hat{x}_i(k+t|k)$. This implies that (3.8) is satisfied, which further guarantees bounded gap between the real state of agent i and the predicted state it sends to its neighbors. This constraint enforces, to a certain extent, consistency between the intended behaviors of an agent and what it makes the neighbors believe how the agent will behave.*

The distributed MPC-based consensus algorithm is formulated in Algorithm 2.

Algorithm 2 Distributed MPC-based Consensus Algorithm

Require: Initial states of all agents x_{i0}; initial assumed state sequences $\hat{x}_i(t|0) = A^t x_{i0}$, $t \in \mathbb{N}_{[0,N]}$ (transmit to neighbors once obtained); index $k = 1$.

Initialization: Every agent i receives the initial assumed state trajectories $\hat{x}_j(t|0)$ from neighbors and solves $\mathcal{P}_i$ without the constraints (3.10f)-(3.10g).

At time instant k,

1: **for** $i \in \mathbb{N}_{[1,M]}$ **do**
2: Agent i receives neighbors optimal predicted state trajectories $x_j^*(k+t-1|k-1)$, $t \in \mathbb{N}_{[0,N]}$, $j \in \mathcal{N}_i$. Then it computes neighbors' assumed state trajectories using (3.11) and stacks them together as $\hat{\boldsymbol{x}}_j(k)$;
3: Agent i measures its current state $x_i(k)$;
4: Solve $\mathcal{P}_i$ to generate optimal predicted control sequences $c_i^*(t+k|k)$ and $u_i^*(t+k|k)$, $t \in \mathbb{N}_{[0,N-1]}$ and obtain optimal predicted state trajectory $x_i^*(t+k|k)$, $t \in \mathbb{N}_{[0,N]}$;
5: Implement the control $u_i(k) = u_i^*(k|k)$ to agent i;
6: Broadcast the optimal predicted state trajectory $x_i^*(t+k|k)$, $t \in \mathbb{N}_{[0,N]}$ to all its neighbors $j \in \mathcal{N}_i$;
7: **end for**
8: Increment k and go back to step 1.

In Initialization of Algorithm 2, solving $\mathcal{P}_i$ without the constraints (3.10f)-(3.10g) is under the assumption that the neighbors apply zero control input over the prediction at the first time step. This idea can be originated from [32].

We compare our distributed MPC-based consensus algorithm with some existing works in the following remark.

Remark 5. *Our algorithm differs some existing works in two aspects. The information broadcasting among our MPC-based consensus controllers is more general, since the communication graph is undirected. In [27], the transmission network is directed, so agent i is able to access its neighbors state $x_j(k)$ simultaneously, thus no assumed state sequences are required to formulate the OCP. Compared with [60] where iterative communication of both state and input information is required, our local MPC controllers exchange state information only once at each time, which relaxes communication load for the network and also saves computational resource for individual local decision makers.*

3.4 Feasibility and Stability Analysis

In the distributed MPC framework, only the first element of the optimal control sequence is implemented to the plant, i.e., $c_i(k) = c_i^*(k|k), u_i(k) = u_i^*(k|k)$. By applying Algorithm 2 to the MAS in (3.1), the closed-loop system becomes

$$x_i(k+1) = Ax_i(k) + BK\{\sum_{j\in\mathcal{N}_i}(x_i(k) - x_j(k)) + w_i(k)\} + Bc_i(k), \tag{3.12}$$

where $w_i(k) := \sum_{j\in\mathcal{N}_i}(x_j(k) - \hat{x}_j(k)) \in \mathcal{W}_i(k)$. The set

$$\mathcal{W}_i(k) := \{w \in \mathbb{R}^n | \, \|w\| \leq \sum_{j\in\mathcal{N}_i} \|x_j(k) - \hat{x}_j(k)\| \leq d_i\Delta_i\}, \tag{3.13}$$

can be easily proved to be compact according to (3.10f). At time instant k, every agent i solves $\mathcal{P}_i$, obtaining the optimal predicted state sequence $x_i^*(k)$, the optimal control sequence $u_i^*(k)$ and the control variable sequence $c_i^*(k)$, where

$$x_i^*(k+t+1|k) = Ax_i^*(k+t|k) + Bu_i^*(k+t|k), \tag{3.14}$$

$$u_i^*(k+t|k) = K\sum_{j\in\mathcal{N}_i}(x_i^*(k+t|k) - \hat{x}_j(k+t|k)) + c_i^*(k+t|k), \quad t \in \mathbb{N}_{[0,N-1]}. \tag{3.15}$$

To approach the cooperation of the overall system, we evaluate the overall closed-loop system

$$\begin{aligned}\boldsymbol{x}(k+1) =&(I_M \otimes A)\boldsymbol{x}(k) + (\mathcal{L} \otimes BK)\boldsymbol{x}(k) + (I_M \otimes BK)\boldsymbol{w}(k) \\ &+ (I_M \otimes B)\boldsymbol{c}(k) \\ =&\varPhi\boldsymbol{x}(k) + (I_M \otimes BK)\boldsymbol{w}(k) + (I_M \otimes B)\boldsymbol{c}(k),\end{aligned} \tag{3.16}$$

where $\varPhi = I_M \otimes A + \mathcal{L} \otimes BK$, $\boldsymbol{w}(k) := \begin{bmatrix} w_1^{\mathrm{T}}(k) & \cdots & w_M^{\mathrm{T}}(k)\end{bmatrix}^{\mathrm{T}} \in \mathcal{W} := \mathcal{W}_1 \times \mathcal{W}_2 \times \cdots \times \mathcal{W}_M$ and $\boldsymbol{c}(k) := \begin{bmatrix} c_1^{\mathrm{T}}(k) & \cdots & c_M^{\mathrm{T}}(k)\end{bmatrix}^{\mathrm{T}}$. By (3.13), one can easily get the compact set $\mathcal{W}(k) = \{\boldsymbol{\omega} \in \mathbb{R}^{n\times M} | \, \|\boldsymbol{\omega}\|^2 \leq \sum_{i=1}^{M} d_i^2 \Delta_i^2\}$.

3.4.1 Feasibility Analysis

In order to prove the recursive feasibility of the distributed MPC-based consensus algorithm by induction, there must exist at least one control sequence $c_i(0)$, such that the constraints (3.10b)-(3.10g) are satisfied for the given initial state x_{i0}. This initial feasibility can be fulfilled by selecting a proper prediction horizon N.

Assume that the initial feasibility is satisfied, the main results on the algorithm feasibility is summarized in the following theorem.

Theorem 3. *For the MAS in (3.1) with initial feasibility, suppose that Assumption 2 and Assumption 3 hold, then Algorithm* 2 *is iteratively feasible.*

Proof. Without loss of generality, we assume that, at an arbitrary time instant k, each agent solves $\mathcal{P}_i$ and the assumed state sequences constructed as in (3.11) are broadcast via the transmission network. By induction principles, the recursive feasibility holds if there exists a control variable candidate sequence $\tilde{c}_i(k+1)$ being compatible to the constraints in $\mathcal{P}_i$ for the state $x_i(k+1)$.

We sketch the proof in two steps. First, we derive the candidate sequences. Concatenating the "tail" of the optimal predicted control variable sequence $c_i^*(k)$ with terminal zero elements, the control variable candidate is formulated as

$$\tilde{c}_i(k+1+t|k+1) = \begin{cases} c_i^*(k+1+t|k), \quad t \in \mathbb{N}_{[0,N-2]}, \\ 0, \quad t = N-1. \end{cases} \tag{3.17}$$

Before showing the candidate $\tilde{c}_i(k+1)$ is compatible for $x_i(k+1)$, we investigate the associated control input candidate $\tilde{u}_i(k+1)$ and the state candidate $\tilde{x}_i(k+1)$ by using (3.14) and (3.15). It is worth mentioning that, at time instant $k+1$, the assumed state trajectories of neighbors are updated to $\hat{x}_j(k+1)$ based on the optimal state trajectories $x_j^*(k)$. Consequently, the control input candidates $\tilde{u}_i(k+1+t|k+1)$ are not necessarily coincide with $u_i^*(k+1+t|k)$ at corresponding time instants, due to the deviation between $\hat{x}_j(k+1+t|k+1)$ and $\hat{x}_j(k+1+t|k)$.

Given the control variable candidate (3.17), the control input and state candidates can be iteratively calculated by

$$\tilde{u}_i(k+1+t|k+1) = \begin{cases} K \sum_{j\in\mathcal{N}_i} (\tilde{x}_i(k+1+t|k+1) - \hat{x}_j(k+1+t|k+1)) \\ \quad + c_i^*(k+1+t|k), \quad t \in \mathbb{N}_{[0,N-2]}, \\ K \sum_{j\in\mathcal{N}_i} (\tilde{x}_i(k+N|k+1) - \hat{x}_j(k+N|k+1)), \quad t = N-1, \end{cases}$$

and

$$\begin{aligned}\tilde{x}_i(k+1+t+1|k+1) =& A\tilde{x}_i(k+1+t|k+1) + B\tilde{u}_i(k+t+1|k+1)\\ =& A\tilde{x}_i(k+1+t|k+1) + BK\sum_{j\in\mathcal{N}_i}(\tilde{x}_i(k+1+t|k+1)\\ &- \hat{x}_j(k+1+t|k+1)) + Bc_i^*(k+1+t|k)\\ =& A\tilde{x}_i(k+1+t|k+1) + BK\sum_{j\in\mathcal{N}_i}(\tilde{x}_i(k+1+t|k+1)\\ &- \hat{x}_j(k+1+t|k))\\ &+ BK\sum_{j\in\mathcal{N}_i}(\hat{x}_j(k+1+t|k) - \hat{x}_j(k+1+t|k+1))\\ &+ Bc_i^*(k+1+t|k), \quad t\in\mathbb{N}_{[0,N-1]},\end{aligned}$$

respectively. Let

$$\begin{aligned}\omega_i^*(k+1+t|k) &= -\sum_{j\in\mathcal{N}_i}(\hat{x}_j(k+1+t|k) - \hat{x}_j(k+1+t|k+1))\\ &= \sum_{j\in\mathcal{N}_i}(x_j^*(k+1+t|k) - \hat{x}_j(k+1+t|k)), \quad t\in\mathbb{N}_{[0,N-1]},\end{aligned}$$

it is easy to check that

$$\|\omega_i^*(k+1+t|k)\| \leq \sum_{j\in\mathcal{N}_i}\|\hat{x}_j(k+1+t|k) - \hat{x}_j(k+1+t|k+1)\| \leq d_i\Delta_i.$$

Using the above expressions, one can represent the state sequence candidate iteratively by

$$\begin{aligned}\tilde{x}_i(k+1+t+1|k+1) =& A\tilde{x}_i(k+1+t|k+1) + BK\sum_{j\in\mathcal{N}_i}(\tilde{x}_i(k+1+t|k+1)\\ &- \hat{x}_j(k+1+t|k)) - BK\omega_i^*(k+1+t|k) + Bc_i^*(k+1+t|k).\end{aligned}$$

Since no uncertainty is involved in the system (3.1), the actual state at $k+1$ can be obtained and represented as follow:

$$x_i(k+1) = Ax_i(k|k) + Bu_i^*(k|k) = x_i^*(k+1|k) = \tilde{x}_i(k+1|k+1),$$

and accordingly, the first element of the control input candidate is given by

$$\begin{aligned}
\tilde{u}_i(k+1|k+1) =& K\sum_{j\in\mathcal{N}_i}(\tilde{x}_i(k+1|k+1) - \hat{x}_j(k+1|k+1)) + \tilde{c}_i(k+1|k+1) \\
=& K\sum_{j\in\mathcal{N}_i}(\tilde{x}_i(k+1|k+1) - \hat{x}_j(k+1|k)) + \tilde{c}_i(k+1|k+1) \\
&+ K\sum_{j\in\mathcal{N}_i}(\hat{x}_j(k+1|k) - \hat{x}_j(k+1|k+1)) \\
=& K\sum_{j\in\mathcal{N}_i}(x_i^*(k+1|k) - \hat{x}_j(k+1|k)) + c_i^*(k+1|k) \\
&- K\sum_{j\in\mathcal{N}_i}(\hat{x}_j(k+1|k+1) - \hat{x}_j(k+1|k)) \\
=& u_i^*(k+1|k) - K\omega_i^*(k+1|k).
\end{aligned}$$

Moreover, it is easy to find the successive element of the state sequence candidate:

$$\begin{aligned}
\tilde{x}_i(k+2|k+1) =& A\tilde{x}_i(k+1|k+1) + B\tilde{u}_i(k+1|k+1) \\
=& Ax_i^*(k+1|k) + B(u_i^*(k+1|k) - K\omega_i^*(k+1|k)) \\
=& x_i^*(k+2|k) - BK\omega_i^*(k+1|k).
\end{aligned}$$

Following the same procedure, the state sequence candidate can be iteratively obtained by

$$\begin{aligned}
\tilde{x}_i(k+1+t|k+1) =& x_i^*(k+1+t|k) \\
&- \sum_{s=0}^{t-1}(A+BK)^{t-1-s}BK\omega_i^*(k+1+s|k), \quad t \geq 1,
\end{aligned} \tag{3.18}$$

and the control sequence candidate as

$$\begin{aligned}\tilde{u}_i(k+t+1|k+1) =& u_i^*(k+1+t|k)\\ &- K\sum_{s=0}^{t-1}(A+BK)^{t-1-s}BK\omega_i^*(k+1+s|k)\\ &- K\omega_i^*(k+1+t|k), \quad t \geq 1,\end{aligned} \tag{3.19}$$

by using the prediction made at k. Though the deviation between prediction and assumed sequences persistently exists, the gap can be restricted within a bounded range thanks to the constraint (3.10f).

We now check the compatibility of the candidates to the constraints in $\mathcal{P}_i$ at $k+1$. Since $\mathcal{P}_i$ is solved at time instant k, it follows that

$$u_i^*(k+t|k) \in \mathcal{U}_i, \quad x_i^*(k+N|k) \in \mathcal{X}_i^f, \quad \forall t \in \mathbb{N}_{[0,N-1]}.$$

In conjunction with (3.18)-(3.19) and the properties of the terminal set, one obtains

$$\begin{aligned}u_i^*(k+1+t|k) =& \tilde{u}_i(k+1+t|k+1) + K\sum_{s=0}^{t-1}(A+BK)^{t-1-s}BK\omega_i^*(k+1+s|k)\\ &+ K\omega_i^*(k+1+t|k) \in \mathcal{U}_i, \quad t \in \mathbb{N}_{[0,N-2]},\end{aligned}$$

We define the variable

$$r_i^t = \begin{cases} 0, \quad t = 0, \\ \sum_{s=0}^{t-1}(A+BK)^{t-1-s}BK\omega_i^*(k+1+s|k), \quad t \in \mathbb{N}_{[1,N-1]} \end{cases}$$

and the set

$$\mathcal{R}_i^t := \bigoplus_{s=0}^{t-1}(A+BK)^{t-1-s}BK\mathcal{W}_i(s), \quad t \geq 1,$$

with $\mathcal{R}_i^0 = \{0\}$. One can get that

$$\begin{aligned} u_i^*(k+1+t|k) =& \tilde{u}_i(k+1+t|k+1) + K r_i^t + K\omega_i^*(k+1+t|k) \\ &\in \tilde{u}_i(k+1+t|k+1) \oplus K\mathcal{R}_i^t \oplus K\mathcal{W}_i(1+t) \in \mathcal{U}_i, \quad t \in \mathbb{N}_{[0,N-2]}. \end{aligned}$$

By the properties of Pontryagin difference, it follows that

$$\tilde{u}_i(k+1+t|k+1) \in ((\mathcal{U}_i \ominus K\mathcal{W}_i(1+t)) \ominus K\mathcal{R}_i^t) \subseteq \mathcal{U}_i, \quad t \in \mathbb{N}_{[0,N-1]}.$$

Therefore, the input constraint is satisfied for the control sequence candidate $\tilde{u}_i(k+1) \in \mathcal{U}_i$. We now check the recursive feasibility for the terminal constraint by plugging

$$\tilde{x}_i(k+N|k+1) = x_i^*(k+N|k) - r_i^{N-1}$$

into

$$\begin{aligned} \tilde{x}_i(k+1+N|k+1) =& A\tilde{x}_i(k+1+N-1|k+1) + B\tilde{u}_i(k+1+N-1|k+1) \\ =& A\tilde{x}_i(k+1+N-1|k+1) + BK\sum_{j\in\mathcal{N}_i}(\tilde{x}_i(k+1+N-1|k+1) \\ &- \hat{x}_j(k+1+N-1|k+1)) + K\underbrace{\tilde{c}_i(k+1+N-1|k+1)}_{\text{zero terminal elements}} \\ =& A\tilde{x}_i(k+1+N-1|k+1) + BK\sum_{j\in\mathcal{N}_i}(\tilde{x}_i(k+1+N-1|k+1) \\ &- \hat{x}_j(k+1+N-1|k)) - BK\sum_{j\in\mathcal{N}_i}(\tilde{x}_j(k+1+N-1|k+1) \\ &- \hat{x}_j(k+1+N-1|k)) \\ =& A\tilde{x}_i(k+1+N-1|k+1) + BK\sum_{j\in\mathcal{N}_i}(\tilde{x}_i(k+1+N-1|k+1) \\ &- \hat{x}_j(k+1+N-1|k)) - BK\omega_i^*(k+1+N-1|k), \end{aligned}$$

then it follows that

$$
\begin{aligned}
\tilde{x}_i(k+1+N|k+1) =& A(x_i^*(k+1+N-1|k) - r_i^{N-1}) \\
& + BK \sum_{j \in \mathcal{N}_i} (x_i^*(k+1+N-1|k) - r_i^{N-1} \\
& - \hat{x}_j(k+1+N-1|k)) - BK\omega_i^*(k+1+N-1|k) \\
=& \tilde{x}_{iN} - (BK\omega_i^*(k+1+N-1|k) + (A+BK)r_i^{N-1}) \\
=& \tilde{x}_{iN} - (BK\omega_i^*(k+1+N-1|k) \\
& + \sum_{s=0}^{N-1-1} (A+BK)^{N-1-s} BK\omega_i^*(k+1+s|k)) \\
=& \tilde{x}_{iN} - \sum_{s=0}^{N-1} (A+BK)^{N-1-s} BK\omega_i^*(k+1+s|k)) \\
=& \tilde{x}_{iN} - \tilde{r}_i^N,
\end{aligned}
$$

where

$$
\tilde{x}_{iN} := \underbrace{Ax_i^*(k+1+N-1|k) + BK \sum_{j \in \mathcal{N}_i} (x_i^*(k+1+N-1|k) - \hat{x}_j(k+1+N-1|k))}_{\text{inside the forward invariant terminal set}}
$$

according to the properties of the forward invariant set [38] and the state deviation

$$
\tilde{b}_i^N = \sum_{s=0}^{N-1} (A+BK)^{N-1-s} BK\omega_i^*(k+1+s|k)) \in \mathcal{R}_i^N.
$$

By the features of the Pontryagin difference, it immediately holds that

$$
\tilde{x}_i(k+1+N|k+1) = \tilde{x}_{iN} - \tilde{b}_i^N \in \mathcal{X}_i^f \ominus \mathcal{R}_i^N \subseteq \mathcal{X}_i^f.
$$

By now, we have shown that, there exists a control variable candidate $\tilde{c}_i(k+1)$ being compatible to the constraints (3.10d), (3.10f), (3.10g) for the successive state $x_i(k+1)$. $\square$

It is worth noting that the consensus control algorithm is executed distributively on local controllers, but consensus describes behaviors of overall systems. In the following two subsections, we investigate the convergence properties of the closed-loop system in (3.16).

3.4.2 Convergence of the Control Variable

Before summarizing the convergence results, we first concatenate the systems in (3.1) into a compact one and restate the preliminaries obtained in the previous subsection.

At time instant k, all agents solve corresponding $\mathcal{P}_i$, obtaining $\boldsymbol{x}^*(k)$, $\boldsymbol{u}^*(k)$ and $\boldsymbol{c}^*(k)$. We immediately get that

$$\begin{aligned}\boldsymbol{x}^*(k+t+1|k) &= (I_M \otimes A)\boldsymbol{x}^*(k+t|k) + (I_M \otimes B)\boldsymbol{u}^*(k+t|k),\\ \boldsymbol{u}^*(k+t|k) &= (\mathcal{L} \otimes K)\boldsymbol{x}^*(k+t|k) + \boldsymbol{c}^*(k+t|k) + (I_M \otimes K)\boldsymbol{\omega}^*(k+t|k).\end{aligned}$$

Following the same line in (3.17), let the control candidate at time instant $k+1$ be

$$\tilde{\boldsymbol{c}}(k+1+t|k+1) = \begin{cases} \boldsymbol{c}^*(k+1+t|k+1), & t \in \mathbb{N}_{[0,N-2]}, \\ 0, \quad t = N-1. \end{cases}$$

Accordingly, the control input candidate and the state candidate are given as follows:

$$\begin{aligned}\tilde{\boldsymbol{u}}(k+1+t|k+1) =& \boldsymbol{u}^*(k+t+1|k) - (I_M \otimes K)\boldsymbol{\omega}^*(k+1+t|k)\\ &- \sum_{s=0}^{t-1} [I_M \otimes (A+BK)^{t-1-s} BK]\boldsymbol{\omega}^*(k+1+s|k)\end{aligned}$$

and

$$
\begin{aligned}
&\tilde{\boldsymbol{x}}(k+1+t+1|k+1)\\
=&(I_M \otimes A)\tilde{\boldsymbol{x}}(k+1+t|k+1) + (I_M \otimes B)\tilde{\boldsymbol{u}}(k+1+t|k+1)\\
=&\boldsymbol{x}^*(k+1+t|k) - \sum_{s=0}^{t-1}(I_M \otimes (A+BK)^{t-1-s}BK)\boldsymbol{\omega}^*(k+1+s|k).
\end{aligned}
$$

Remark 6. *Due to the existence of "errors" (treated as disturbance) between the actual state and assumed state $x_i(k)-\hat{x}_i(k)$, the overall system in* (3.16) *is not possible to guarantee asymptotic stability with respect to the consensus set. We however can steer* (3.16) *to a neighborhood set of $\mathcal{C}$, which further translates to the neighborhood set $\mathcal{C} \oplus \mathcal{R}$ being as small as possible. By tightening the set $\mathcal{R}$, the state sensitivity to the state deviation is minimized.*

The convergence of the control variable is summarized in the following lemma.

Lemma 8. *Given any feasible initial state $\boldsymbol{x}_0$, the optimal control sequences $\lim_{k\to\infty} \boldsymbol{c}^*(k) = 0$ and square summable, i.e. $\lim_{k\to\infty} \sum_{t=0}^{k}(\boldsymbol{c}^*(t))^{\mathrm{T}}\boldsymbol{c}^*(t) < \infty$.*

Proof. Denote the Lyapunov function by

$$V_C(k) := J^*(\boldsymbol{c}(k)) = \sum_{i=1}^{M} J_N^i(k) = \sum_{t=0}^{N-1} \|\boldsymbol{c}^*(k+t|k)\|^2 \geq 0.$$

For the control candidate sequence $\tilde{\boldsymbol{c}}(k+1)$, we have

$$\tilde{V}_C(k+1) = \sum_{t=0}^{N-1} \|\tilde{\boldsymbol{c}}(k+1+t|k+1)\|^2 = V_C(k) - \|\boldsymbol{c}^*(k|k)\|^2.$$

Since the control variable candidate $\tilde{\boldsymbol{c}}(k+1)$ is a feasible, but not necessarily an optimal solution to $\mathcal{P}_i$, the associated cost value is suboptimal to the OCP at time instant $k+1$, so it holds that

$$\begin{aligned}\tilde{V}_C(k+1) &= V_C(k) - \left\|\boldsymbol{c}^*(k|k)\right\|^2 \leq V_C(k+1),\\ \Rightarrow &V_C(k+1) - V_C(k) \leq -\left\|\boldsymbol{c}^*(k|k)\right\|^2 \leq 0.\end{aligned} \tag{3.20}$$

Considering the sum of the inequalities in (3.20), one gets

$$\lim_{k\to\infty}\sum_{t=0}^{k}(V_C(t+1) - V_C(t)) = \lim_{k\to\infty} V_C(k+1) - V_C(0) \leq -\lim_{k\to\infty}\sum_{t=0}^{k}\left\|\boldsymbol{c}^*(t|t)\right\|^2.$$

As $V_C(k+1)$ and $V_C(0)$ are sums of squared finite scalars, we obtain that

$$0 \leq \lim_{k\to\infty} V_C(k+1) = V_C(0) - \lim_{k\to\infty}\sum_{t=0}^{k}\left\|\boldsymbol{c}^*(t|t)\right\|^2 < \infty,$$

which implies $\lim\limits_{k\to\infty}\sum_{t=0}^{k}\left\|\boldsymbol{c}^*(t|t)\right\|^2 < \infty$. Furthermore, by the properties of square summable infinite series, it follows that

$$\lim_{k\to\infty}\left\|\boldsymbol{c}^*(k|k)\right\|^2 = 0$$

and each entry of the control variable vector $\boldsymbol{c}(k)$ also satisfies $\lim\limits_{k\to\infty} c_i^*(k) = \lim\limits_{k\to\infty} c_i^*(k|k) = 0$. By now we prove the convergence of the control variable $c_i(k)$ as it vanishes to zero as $k\to\infty$. □

3.4.3 Consensus Convergence Analysis

Now we consider the convergence of the MAS in (3.16). Due to the consistent deviation between the real current neighbor states and assumed ones, it can be checked that the evolution of the actual system state at time instant k is given by

$$\begin{aligned} \boldsymbol{x}(k) =& \Phi^k \boldsymbol{x}(0) + \sum_{t=0}^{k-1} \Phi^t (I_M \otimes B) \boldsymbol{c}(k-1-t) \\ &+ \sum_{t=0}^{k-1} \Phi^t (I_M \otimes BK) \boldsymbol{w}(k-1-t), \end{aligned} \tag{3.21}$$

where $\boldsymbol{c}(k-1-t) = \boldsymbol{c}^*(k-1-t|k-1-t)$, $\boldsymbol{w}(k-1-t) = \boldsymbol{w}^*(k-1-t|k-1-t)$. The corresponding average state of the MAS is defined by

$$\bar{\boldsymbol{x}}(k) = \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M) \otimes I_n)\boldsymbol{x}(k) = \begin{bmatrix} \bar{x}(k)^{\mathrm{T}} & \cdots & \bar{x}(k)^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \in \mathcal{C}, \tag{3.22}$$

where $\bar{x}(k) = 1/M \sum_{i=1}^{M} x_i(k)$. It is obvious that $\bar{x}(k)$ in the average state vector $\bar{\boldsymbol{x}}(k)$ are equal, which implies $\bar{\boldsymbol{x}}(k) \in \mathcal{C}$. From the perspective of set theory,

$$\lim_{k\to\infty} |\boldsymbol{x}(k)|_{\mathcal{C}} = \lim_{k\to\infty} \|\boldsymbol{x}(k) - \bar{\boldsymbol{x}}(k)\|$$

measures the distance from the actual state to the consensus set. We investigate the distance between the actual state (3.21) and the average state (3.22) as $k \to \infty$:

$$\begin{aligned} &\lim_{k\to\infty} \boldsymbol{x}(k) - \bar{\boldsymbol{x}}(k) \\ =& \lim_{k\to\infty} \left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M) \otimes I_n) \right] \boldsymbol{x}(k) \\ =& \lim_{k\to\infty} \left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M) \otimes I_n) \right] \Phi^k \boldsymbol{x}(0) \\ &+ \lim_{k\to\infty} \sum_{t=0}^{k-1} \left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M) \otimes I_n) \right] \Phi^t (I_M \otimes B) \boldsymbol{c}(k-1-t) \\ &+ \lim_{k\to\infty} \sum_{t=0}^{k-1} \left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M) \otimes I_n) \right] \Phi^t (I_M \otimes BK) \boldsymbol{w}(k-1-t). \end{aligned} \tag{3.23}$$

Before proving that the first two terms in (3.23) vanish to zero as $k \to \infty$, we first recall an useful lemma from [78].

Lemma 9 ([78]). *For any given scalar* $\beta \in (0,1)$, *and square summable series* $\lim_{k\to\infty} E(k)^2 < \infty$, *it holds that*

$$\lim_{k\to\infty} \sum_{t=0}^{k-1} \beta^{k-1-t} E(t) = 0.$$

Now we prove the following lemma by using Lemma 9.

Lemma 10. *For the transition matrix* $\Phi = I_M \otimes A + \mathcal{L} \otimes BK$ *of the MAS in (3.16) consisting of semi-stable agents, it holds that*

$$\lim_{k\to\infty} \left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}} \mathbf{1}_M) \otimes I_n) \right] \Phi^k \boldsymbol{x}(0) = 0.$$

Proof. By application of the semi-stability theorem in [74] and the properties of Drazin inverse, we have

$$\begin{aligned} Y = &\lim_{k\to\infty} \left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}} \mathbf{1}_M) \otimes I_n) \right] \Phi^k \\ =&\left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}} \mathbf{1}_M) \otimes I_n) \right] \left[I_{Mn} - (I_n - \Phi)(I_n - \Phi)^{\#} \right]. \end{aligned}$$

We further obtain that

$$\begin{aligned} Y(I_{Mn} - \Phi) =&\Big[I_{Mn} - (\Phi - I_{Mn})(\Phi - I_{Mn})^{\#} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}} \mathbf{1}_M) \otimes I_n) \\ &+ \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}} \mathbf{1}_M) \otimes I_n)(\Phi - I_{Mn})(\Phi - I_{Mn})^{\#} \Big] (I_{Mn} - \Phi) \\ =&(I_{Mn} - \Phi) - (I_{Mn} - \Phi) - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}} \mathbf{1}_M) \otimes I_n)(I_{Mn} - \Phi) \\ &+ \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}} \mathbf{1}_M) \otimes I_n)(I_{Mn} - \Phi) = 0, \end{aligned} \tag{3.24}$$

which implies $Y = 0$. Therefore we prove that

$$\lim_{k\to\infty}\left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]\Phi^k\boldsymbol{x}(0) = \lim_{k\to\infty} Y\boldsymbol{x}(0) = 0.$$

□

Now we give the proof of the convergence of the second term in (3.23) in the following lemma.

Lemma 11. *Consider the second term in (3.23), it holds that*

$$\lim_{k\to\infty}\sum_{t=0}^{k-1}\left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]\Phi^t(I_M\otimes B)\boldsymbol{c}(k-1-t) = 0. \tag{3.25}$$

Proof. Since (3.24) holds, there always exists a scalar $0 < \beta < 1$, such that

$$\lambda_{\max}(\left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]\Phi^t) \leq \lambda_{\max}(\left[I_{Mn} - \frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]) \leq \beta^t < 1.$$

Considering that the control variable sequence $\boldsymbol{c}^*(k)$ is square summable, it also holds that the norm sequence $E(k-1-t) = \|I_M\otimes B\|\,\|\boldsymbol{c}(k-1-t)\|$ is square summable:

$$\lim_{k\to\infty}\sum_{t=0}^{k-1}E^2(k-1-t) = \|I_M\otimes B\|^2\lim_{k\to\infty}\sum_{t=0}^{k-1}\|\boldsymbol{c}(k-1-t)\|^2 < \infty.$$

By application of the Cauchy-Schwarz inequality, the Euclidean norm of the sum in

(3.25) becomes

$$
\begin{aligned}
0 &\le \lim_{k\to\infty}\left\|\sum_{t=0}^{k-1}\left[I_{Mn}-\frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]\Phi^t(I_M\otimes B)\boldsymbol{c}(k-1-t)\right\| \\
&\le \lim_{k\to\infty}\sum_{t=0}^{k-1}\left\|\left[I_{Mn}-\frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]\Phi^t\right\|\|(I_M\otimes B)\boldsymbol{c}(k-1-t)\| \\
&\le \lim_{k\to\infty}\sum_{t=0}^{k-1}\beta^t E(k-1-t).
\end{aligned}
$$

In conjunction with Lemma 9, we immediately obtain

$$
\lim_{k\to\infty}\sum_{t=0}^{k-1}\beta^t E(k-1-t)=0,
$$

implying that

$$
\lim_{k\to\infty}\left\|\sum_{t=0}^{k-1}\left[I_{Mn}-\frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]\Phi^t(I_M\otimes B)\boldsymbol{c}(k-1-t)\right\|=0. \tag{3.26}
$$

□

Now we are in a position to summarize our consensus convergence results. The results in (3.25) and (3.26) reveal that the overall state will converge to a bounded invariant set around the consensus set $\mathcal{C}$. Together with (3.23), we can conclude that

$$
\begin{aligned}
\lim_{k\to\infty}\boldsymbol{x}(k) &= \bar{\boldsymbol{x}}(k)+\lim_{k\to\infty}\sum_{t=0}^{k-1}\left[I_{Mn}-\frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]\Phi^t(I_M\otimes BK)\boldsymbol{w}(k-1-t) \\
&= \bar{\boldsymbol{x}}(k)+\lim_{k\to\infty}\sum_{t=0}^{k-1}\bar{\Phi}^t(I_M\otimes BK)\boldsymbol{w}(k-1-t),
\end{aligned}
$$

where matrix $\bar{\Phi}$ is Hurwitz and satisfies $\bar{\Phi}^t=\left[I_{Mn}-\frac{1}{M}((\mathbf{1}_M^{\mathrm{T}}\mathbf{1}_M)\otimes I_n)\right]\Phi^t$. Similarly, we use some results of the robust positively invariant set presented in [71]. Let

$$\mathcal{R}_t = \bigoplus_{s=0}^{t-1} \bar{\Phi}^s (I_M \otimes BK)\mathcal{W}$$

be the deviation set of states which is reachable in j steps starting from the consensus set $\mathcal{C}$. It is obvious that $\sum_{t=0}^{k-1} \bar{\Phi}^t (I_M \otimes BK)\boldsymbol{w}(k-1-t) \in \mathcal{R}_k$. By [71], the set $\mathcal{R}_t$ remains bounded as $t \to \infty$ and contains the origin as an interior point, so we take the compact set $\mathcal{R} := \mathcal{R}_\infty$. Therefore, under the initial feasibility, the overall state of the MAS guarantees the convergence to a neighborhood of the consensus set, i.e.,

$$\lim_{k\to\infty} \boldsymbol{x}(k) \in \mathcal{C} \oplus \lim_{k\to\infty} \mathcal{R}_k = \mathcal{C} \oplus \mathcal{R}.$$

3.5 Numerical Examples

Consider an MAS with 5 identical linear semi-stable dynamics given by

$$x_i(k+1) = Ax_i(k) + Bu_i(k), \quad x_i(0) = x_{i0}$$

where

$$A = \begin{bmatrix} 0.8 & 0.1 & 0.1 & 0 & 0 \\ 0 & 0.9 & 0 & 0.1 & 0 \\ 0.1 & 0.1 & 0.6 & 0.1 & 0.1 \\ 0 & 0.1 & 0.1 & 0.8 & 0 \\ 0.1 & 0.1 & 0 & 0 & 0.8 \end{bmatrix}, \quad B = \begin{bmatrix} -0.1 & 0.1 \\ 0.1 & -0.2 \\ 0 & -0.3 \\ 0.08 & 0.1 \\ 0.2 & 0.08 \end{bmatrix}.$$

The initial state of the five agents are selected as follow:

$$x_{10} = \begin{bmatrix} 0.9442 & 1.2174 & -1.1190 & 1.2401 & 0.3971 \end{bmatrix}^{\mathrm{T}}$$
$$x_{20} = \begin{bmatrix} 1.2074 & -0.6645 & 0.1406 & 1.3725 & 1.3947 \end{bmatrix}^{\mathrm{T}}$$
$$x_{30} = \begin{bmatrix} -1.0272 & 1.4118 & 1.3715 & -0.0439 & 0.9008 \end{bmatrix}^{\mathrm{T}},$$
$$x_{40} = \begin{bmatrix} -1.0743 & -0.2347 & 1.2472 & 0.8766 & 1.3785 \end{bmatrix}^{\mathrm{T}},$$
$$x_{50} = \begin{bmatrix} 0.4672 & -1.3929 & 1.0474 & 1.3020 & 0.5362 \end{bmatrix}^{\mathrm{T}}.$$

The control input for every agent is restricted by $\|u_i\|_\infty \leq 0.3$. The communication network underlying the MAS contains a spanning tree and its Laplacian is

$$\mathcal{L} = \begin{bmatrix} 2 & -1 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & 0 & -1 \\ -1 & 0 & 0 & 2 & -1 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix}.$$

The parameters are designed based on the inverse optimal consensus method in [38] as follows: $S_2 = [2.551, -0.447, 0.119, -0.813, -1.069; -0.447, 4.028, 0.227, 1.356, -2.664; 0.119, 0.227, 1.799, 0.74, -2.431, -0.813, 1.356, 0.74, 3.884, -3.689, -1.069, -2.664, -2.431, -3.689, 10.081]$, $K = [0.1258, -0.1015, 0.0542, 0.0071, -2.443; -0.0787, 0.1863, 0.0637, 0.0982, -0.1376]$. The simulation results of the pre-stabilizing linear consensus protocol in (3.4) are shown in Figure 3.1 and Figure 3.2. From Figure 3.1, we can see that the corresponding states of all five agents converge to the same value and the closed-loop system reaches consensus under the design pre-stabilizing consensus control. However, in Figure 3.2, it can be seen that the control inputs of some agents exceed the restriction of 0.3.

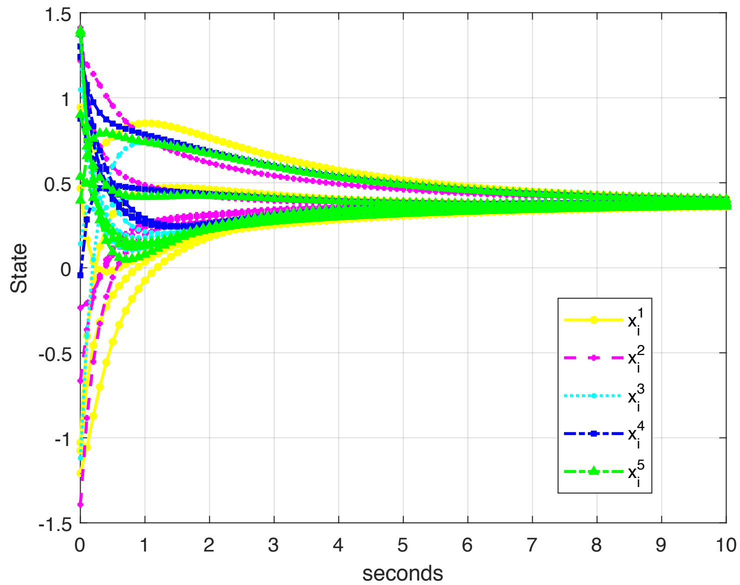
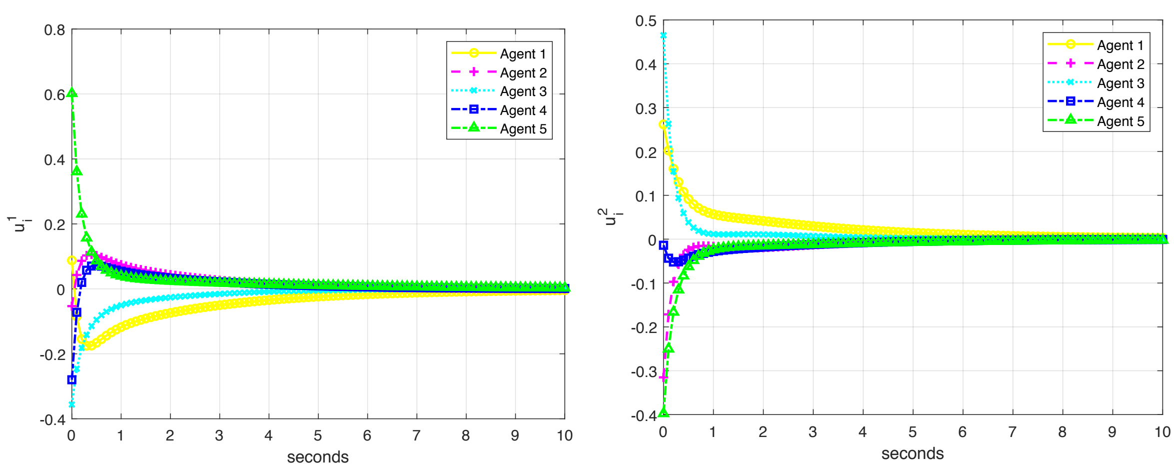


Figure 3.1: State trajectories of all agents



(a) Pre-stabilizing control inputs for u_i^1 coordinates

(b) Pre-stabilizing control inputs for u_i^2 coordinates

Figure 3.2: Pre-stabilizing control inputs for all 5 agents

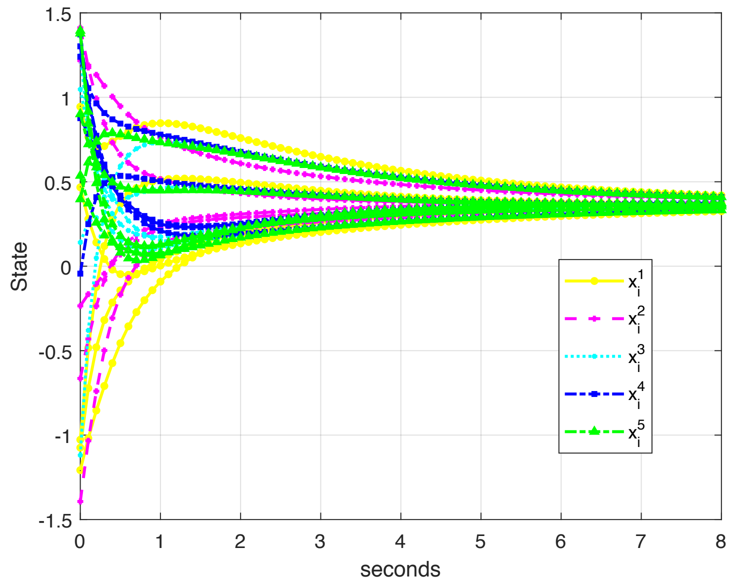
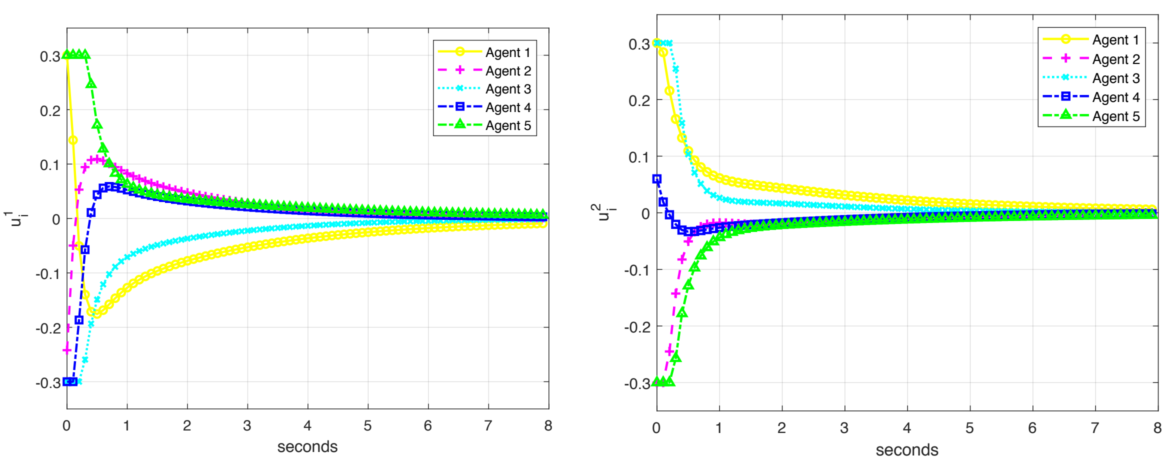


Figure 3.3: State trajectories of all agents



(a) MPC inputs for u_i^1 coordinates

(b) MPC inputs for u_i^2 coordinates

Figure 3.4: MPC inputs for all 5 agents

In the distributed MPC-based consensus framework setup, the prediction horizon is taken as $N = 10$, and $\Delta = 0.3$, $\beta_s = 13.9$. Making use of the MATLAB packages, YALMIP and MPT, we obtain the simulation results as shown in the above figures.

It is shown in Figure 3.3 that all states of every agent converge to a small neighboring area around 0. This indicates that the closed-loop MAS converge to a neighboring set of the consensus set under the proposed distributed MPC-based consensus algorithm. The control inputs for all agents can be seen in Figure 3.4. The figures imply that the input constraints are satisfied and the proposed distributed MPC framework meets the control objectives.

3.6 Conclusion

In this chapter, the MPC-based consensus problem for input-constrained linear MASs is studied. An inverse optimal consensus feedback is first introduced as the pre-stabilizing consensus protocol in unconstrained scenarios. The next step is to design the distributed MPC-based consensus algorithm based on the pre-stabilizing control law with the satisfaction of the input constraints. Moreover, the recursive feasibility of the proposed distributed MPC scheme has been analyzed. The overall system is regulated to a neighborhood of the consensus set due to the distributed nature of the digital controller networks.

Chapter 4

Conclusions and Future Work

4.1 Conclusions

In this thesis, the MPC-based consensus problem for linear MASs has been addressed.

Chapter 2 involves a centralized tube-based MPC scheme to solve the consensus problem for constrained linear MASs with bounded additive disturbance. First, a linear consensus control protocol is designed via a suboptimal, rather than optimal, linear quadratic approach, due to the non-convexity of the associated consensus performance function. With the linear consensus protocol, we extend an existing robust MPC scheme for single system stabilization to solve the multi-agent consensus problem. In order to robustify the model predictive controller with respect to persistent disturbance, proper restrictions on the constraints are implemented. In this way, with feasible initial states, the iterative feasibility of the proposed MPC scheme can be guaranteed. As the robust constraint sets can be computed offline, no extra online computation is required.

In Chapter 3, we propose a novel distributed MPC-based consensus protocol for linear MASs with semi-stable dynamics. In order to reach consensus and guarantee recursive feasibility, each agent optimizes a local cost function at every sampling time to obtain the optimal predicted control sequence and state sequence. The predicted information is shared with neighbors via a communication network with fixed topology. Moreover, an inverse optimal linear control serves as the pre-stabilizing consensus protocol for the proposed distributed MPC scheme.

4.2 Future Work

In this thesis, some assumptions are made to formulate and solve the consensus problem by using MPC schemes. These assumptions may restrict practical implementation of the proposed control algorithms. Many interesting areas and problems are worth of investigating in the future. We list some research potentials here.

- Chapter 2 involves the centralized consensus control problem of multi-agent systems. The centralized control scheme can achieve the desired consensus performance if the central controller is computationally powerful enough and the scale of the MASs is not very large. However, in most real applications, controllers cannot provide such powerful computational resources when the number of agents keeps increasing. Generally speaking, decoupling the large scale systems into single ones and then assigning each agent an independent local controller can significantly reduce the computational load. Decentralized control schemes are quite promising potential solutions to the consensus control of MASs, but the design of such decentralized algorithms is still very challenging.

- Another potential research branch would be how to address the consensus problem for general nonlinear MASs with bounded disturbance in a distributed way as in Chapter 3. The consensus control strategies designed via distributed MPC-based consensus protocol for general linear time-invariant MASs with input constraints, avoiding iterative communication among subsystems and reducing sequential information transmission via the networks, as well as in a fully distributed manner is still difficult to solve and only few existing works can be found.

Appendix A

Additional Information

Bibliography

[1] L. Brinón-Arranz, A. Seuret, and C. Canudas-de Wit, "Cooperative control design for time-varying formations of multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 8, pp. 2283–2288, 2014.

[2] Z. Lin, L. Wang, Z. Han, and M. Fu, "Distributed formation control of multi-agent systems using complex Laplacian," *IEEE Transactions on Automatic Control*, vol. 59, no. 7, pp. 1765–1777, 2014.

[3] J. Zhan and X. Li, "Flocking of multi-agent systems via model predictive control based on position-only measurements," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 377–385, 2012.

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

[5] Z. Li, H. H. Liu, B. Zhu, and H. Gao, "Robust second-order consensus tracking of multiple 3-DOF laboratory helicopters via output feedback," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2538–2549, 2015.

[6] Y.-C. Choi and H.-S. Ahn, "Nonlinear control of quadrotor for point tracking: Actual implementation and experimental tests," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 3, pp. 1179–1192, 2014.

[7] Y.-C. Ho, "On centralized optimal control," *IEEE Transactions on Automatic Control*, vol. 50, no. 4, pp. 537–538, 2005.

[8] W. Wang, D. E. Rivera, and K. G. Kempf, "Centralized model predictive control strategies for inventory management in semiconductor manufacturing supply chains," in *Proceedings of the American Control Conference*, vol. 1, pp. 585–590, 2003.

[9] D. Raimondo, L. Magni, and R. Scattolini, "Decentralized MPC of nonlinear systems: An input-to-state stability approach," *International Journal of Robust and Nonlinear Control*, vol. 17, no. 17, pp. 1651–1667, 2007.

[10] J. B. Rawlings and B. T. Stewart, "Coordinating multiple optimization-based controllers: New opportunities and challenges," *Journal of Process Control*, vol. 18, no. 9, pp. 839–845, 2008.

[11] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia, "Cooperative distributed model predictive control," *Systems & Control Letters*, vol. 59, no. 8, pp. 460–469, 2010.

[12] D. Jia and B. Krogh, "Min-max feedback model predictive control for distributed control with communication," in *Proceedings of the American Control Conference*, vol. 6, pp. 4507–4512, 2002.

[13] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE Control Systems Magazine*, vol. 22, no. 1, pp. 44–52, 2002.

[14] W. Ren, R. W. Beard, and E. M. Atkins, "A survey of consensus problems in multi-agent coordination," in *Proceedings of American Control Conference*, pp. 1859–1864 vol. 3, 2005.

[15] N. A. Lynch, *Distributed algorithms.* Elsevier, 1996.

[16] Wei Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, 2005.

[17] K. You and L. Xie, "Network topology and communication data rate for consensusability of discrete-time multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2262–2275, 2011.

[18] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *Departmental Papers (ESE)*, p. 29, 2003.

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

[20] M. Cao, A. S. Morse, and B. D. Anderson, "Reaching a consensus in a dynamically changing environment: Convergence rates, measurement delays, and asynchronous events," *SIAM Journal on Control and Optimization*, vol. 47, no. 2, pp. 601–623, 2008.

[21] W. Ren and R. W. Beard, *Consensus algorithms for double-integrator dynamics.* Springer, 2008.

[22] G. Ferrari-Trecate, L. Galbusera, M. P. E. Marciandi, and R. Scattolini, "Model predictive control schemes for consensus in multi-agent systems with single-and double-integrator dynamics," *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2560–2572, 2009.

[23] B. Zhu, K. Guo, and L. Xie, "A new distributed model predictive control for unconstrained double-integrator multiagent systems," *IEEE Transactions on Automatic Control*, vol. 63, no. 12, pp. 4367–4374, 2018.

[24] W. Yu, G. Chen, and M. Cao, "Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems," *Automatica*, vol. 46, no. 6, pp. 1089–1095, 2010.

[25] H. Su, M. Z. Chen, J. Lam, and Z. Lin, "Semi-global leader-following consensus of linear multi-agent systems with input saturation via low gain feedback," *IEEE Transactions on Circuits and Systems*, vol. 60, no. 7, pp. 1881–1889, 2013.

[26] Y. Kuriki and T. Namerikawa, "Consensus-based cooperative formation control with collision avoidance for a multi-UAV system," in *Proceedings of American Control Conference*, pp. 2077–2082, 2014.

[27] J. Zhan, Z.-P. Jiang, Y. Wang, and X. Li, "Distributed model predictive consensus with self-triggered mechanism in general linear multi-agent systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 3987–3997, 2018.

[28] J. Jiao, H. L. Trentelman, and M. K. Camlibel, "A suboptimality approach to distributed linear quadratic optimal control," *IEEE Transactions on Automatic Control*, vol. 14, no. 8, 2019.

[29] S. Manfredi and D. Angeli, "Asymptotic consensus on the average of a field for time-varying nonlinear networks under almost periodic connectivity," *IEEE Transactions on Automatic Control*, vol. 63, no. 8, pp. 2389–2404, 2017.

[30] A. Nikou and D. V. Dimarogonas, "Decentralized tube-based model predictive control of uncertain nonlinear multiagent systems," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 10, pp. 2799–2818, 2019.

[31] R. W. Beard and T. W. McLain, "Multiple UAV cooperative search under collision avoidance and limited range communication constraints," in *Proceedings of the 42nd IEEE International Conference on Decision and Control*, vol. 1, pp. 25–30, IEEE, 2003.

[32] W. B. Dunbar and R. M. Murray, "Distributed receding horizon control for multi-vehicle formation stabilization," *Automatica*, vol. 42, no. 4, pp. 549–558, 2006.

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

[34] S. Kar and J. M. Moura, "Distributed consensus algorithms in sensor networks: Quantized data and random link failures," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1383–1400, 2009.

[35] J. M. Solanki, S. Khushalani, and N. N. Schulz, "A multi-agent solution to distribution systems restoration," *IEEE Transactions on Power Systems*, vol. 22, no. 3, pp. 1026–1034, 2007.

[36] J. Alejandro, A. Arce, and C. Bordons, "An integrated framework for distributed model predictive control of large-scale power networks," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 197–209, 2013.

[37] A. Nikou and D. V. Dimarogonas, "Decentralized tube-based model predictive control of uncertain nonlinear multiagent systems," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 10, pp. 2799–2818, 2019.

[38] H. Li, Y. Shi, W. Yan, and F. Liu, "Receding horizon consensus of general linear multi-agent systems with input constraints: An inverse optimality approach," *Automatica*, vol. 91, pp. 10–16, 2018.

[39] M. Farina and R. Scattolini, "Distributed predictive control: A non-cooperative algorithm with neighbor-to-neighbor communication for linear systems," *Automatica*, vol. 48, no. 6, pp. 1088–1096, 2012.

[40] M. Farina and R. Scattolini, "Distributed non-cooperative MPC with neighbor-to-neighbor communication," *in IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 404–409, 2011.

[41] C. Shen, Y. Shi, and B. Buckham, "Integrated path planning and tracking control of an AUV: A unified receding horizon optimization approach," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 3, pp. 1163–1173, 2016.

[42] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[43] H. Chen and F. Allgöwer, "A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability," *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.

[44] D. L. Marruedo, T. Alamo, and E. Camacho, "Input-to-state stable MPC for constrained discrete-time nonlinear systems with bounded additive uncertainties," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 4, pp. 4619–4624, 2002.

[45] L. Chisci, J. A. Rossiter, and G. Zappa, "Systems with persistent disturbances: predictive control with restricted constraints," *Automatica*, vol. 37, no. 7, pp. 1019–1028, 2001.

[46] D. Mayne, M. Seron, and S. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[47] P. O. Scokaert and D. Q. Mayne, "Min-max feedback model predictive control for constrained linear systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 8, pp. 1136–1142, 1998.

[48] D. M. Raimondo, D. Limon, M. Lazar, L. Magni, and E. F. ndez Camacho, "Min-max model predictive control of nonlinear systems: A unifying overview on stability," *European Journal of Control*, vol. 15, no. 1, pp. 5–21, 2009.

[49] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.

[50] L. Magni, G. De Nicolao, R. Scattolini, and F. Allgöwer, "Robust model predictive control for nonlinear discrete-time systems," *International Journal of Robust and Nonlinear Control*, vol. 13, no. 3-4, pp. 229–246, 2003.

[51] J. Schuurmans and J. Rossiter, "Robust predictive control using tight sets of predicted states," *Control Theory and Applications*, vol. 147, no. 1, pp. 13–18, 2000.

[52] W. B. Dunbar, "Distributed receding horizon control of dynamically coupled nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1249–1263, 2007.

[53] C. Wang and C.-J. Ong, “Distributed model predictive control of dynamically decoupled systems with coupled cost,” *Automatica*, vol. 46, no. 12, pp. 2053–2058, 2010.

[54] G. Ferrari-Trecate, L. Galbusera, M. P. E. Marciandi, and R. Scattolini, “Model predictive control schemes for consensus in multi-agent systems with single-and double-integrator dynamics,” *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2560–2572, 2009.

[55] H. Li and Y. Shi, “Robust distributed model predictive control of constrained continuous-time nonlinear systems: A robustness constraint approach,” *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1673–1678, 2014.

[56] Z. Cheng, H. Zhang, M. Fan, and G. Chen, “Distributed consensus of multi-agent systems with input constraints: A model predictive control approach,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, pp. 825–834, March 2015.

[57] J. Zhan and X. Li, “Consensus of sampled-data multi-agent networking systems via model predictive control,” *Automatica*, vol. 49, no. 8, pp. 2502–2507, 2013.

[58] H. Li and W. Yan, “Receding horizon control based consensus scheme in general linear multi-agent systems,” *Automatica*, vol. 56, pp. 12–18, 2015.

[59] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, “Distributed event-triggered control for multi-agent systems,” *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1291–1297, 2011.

[60] M. A. Müller, B. Schürmann, and F. Allgöwer, “Robust cooperative control of dynamically decoupled systems via distributed MPC,” *in IFAC Proceedings Volumes*, vol. 45, no. 17, pp. 412–417, 2012.

[61] H. Li, Y. Shi, and W. Yan, "On neighbor information utilization in distributed receding horizon control for consensus-seeking," *IEEE Transactions on Cybernetics*, vol. 46, no. 9, pp. 2019–2027, 2015.

[62] P.-A. Bliman and G. Ferrari-Trecate, "Average consensus problems in networks of agents with delayed communications," *Automatica*, vol. 44, no. 8, pp. 1985–1995, 2008.

[63] Y. Chen and Y. Shi, "Consensus for linear multiagent systems with time-varying delays: A frequency domain perspective," *IEEE Transactions on Cybernetics*, vol. 47, no. 8, pp. 2143–2150, 2016.

[64] J. Qin, H. Gao, and W. X. Zheng, "Second-order consensus for multi-agent systems with switching topology and communication delay," *Systems & Control Letters*, vol. 60, no. 6, pp. 390–397, 2011.

[65] J. Qin, C. Yu, and H. Gao, "Coordination for linear multiagent systems with dynamic interaction topology in the leader-following framework," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 5, pp. 2412–2422, 2013.

[66] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[67] J. Rawlings and D. Mayne, "Postface to model predictive control: Theory and design," *Nob Hill Pub*, vol. 5, pp. 155–158, 2012.

[68] J. M. Maciejowski, *Predictive control: With constraints.* Pearson Education, 2002.

[69] L. Grüne and J. Pannek, *Nonlinear model predictive control.* Springer, 2017.

[70] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design.* Nob Hill Pub., 2009.

[71] I. Kolmanovsky and E. G. Gilbert, “Maximal output admissible sets for discrete-time systems with disturbance inputs,” in *Proceedings of American Control Conference*, vol. 3, 1995.

[72] L. Moreau, “Stability of continuous-time distributed consensus algorithms,” in *Proceedings of the 43rd IEEE Conference on Decision and Control*, vol. 4, pp. 3998–4003, 2004.

[73] V. Fragoso-Rubio, M. Velasco-Villa, M. Vallejo-Alarcón, J. Vásquez-Santacruz, and M. Hernández-Pérez, “Consensus problem for linear time-invariant systems with time-delay,” *Mathematical Problems in Engineering*, vol. 2019, 2019.

[74] Q. Hui and W. M. Haddad, “Optimal semistable stabilisation for linear discrete-time dynamical systems with applications to network consensus,” *International Journal of Control*, vol. 82, no. 3, pp. 456–469, 2009.

[75] Q. Hui and Z. Liu, “Semistability-based robust and optimal control design for network systems,” in *Proceedings of the 51st IEEE Conference on Decision and Control*, pp. 7049–7054, 2012.

[76] S. Zeng and F. Allgöwer, “Structured optimal feedback in multi-agent systems: A static output feedback perspective,” *Automatica*, vol. 76, pp. 214–221, 2017.

[77] Z.-P. Jiang and Y. Wang, “A converse Lyapunov theorem for discrete-time systems with disturbances,” *Systems & Control Letters*, vol. 45, no. 1, pp. 49–58, 2002.

[78] C. Liu, H. Li, Y. Shi, and D. Xu, "Distributed event-triggered gradient method for constrained convex minimization," *IEEE Transactions on Automatic Control*, 2019.