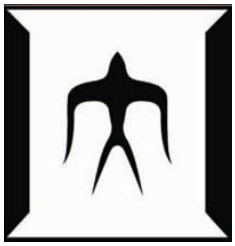
論文 / 著書情報
Article / Book Information

題目(和文)	
Title(English)	Resilient Consensus in Multi-Agent Systems with Limited Resources
著者(和文)	WangYuan
Author(English)	Yuan Wang
出典(和文)	学位:博士(学術), 学位授与機関:東京工業大学, 報告番号:甲第11334号, 授与年月日:2019年9月20日, 学位の種別:課程博士, 審査員:石井 秀明,山村 雅幸,三宅 美博,DEFAGO XAVIER,小野 功
Citation(English)	Degree:Doctor (Academic), Conferring organization: Tokyo Institute of Technology, Report number:甲第11334号, Conferred date:2019/9/20, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

Resilient Consensus in Multi-Agent Systems with Limited Resources



Yuan Wang
Department of Computer Science
Tokyo Institute of Technology

Supervisor
Hideaki Ishii

In partial fulfillment of the requirements for the degree of
Doctor of Philosophy
July, 2019

Abstract

In large-scale multi-agent systems, consensus problems form one of the fundamental problems related to distributed algorithms. There, agents interact locally and exchange their information with each other in order to arrive at the global objective of sharing a common value. In recent years, security problems in multi-agent systems have become a critical issue. Malicious attacks can lead the systems to undesirable operations or even accidents. In an uncertain environment where faults or even adversarial attacks can be present, it is of great importance to defend consensus algorithms by raising their security levels so as to avoid being influenced by such uncertainties in their decision makings.

This thesis studies the problem of resilient consensus in multi-agent systems where attackers may update the adversary agents value arbitrarily. The objective of adversary agents is to prevent the regular agents from reaching consensus. We focus on solving the resilient consensus problem with emphasis on resource saving. In particular, we study the saving of four resources: (1) Communication resources, (2) memory resources, (3) energy resources, and (4) graph resources. The thesis consists of four parts dealing with these issues as follows:

(1) We consider resilient versions of discrete-time multi-agent consensus in the presence of faulty or even malicious agents in the network.

To save communication resources, we develop event-triggered update rules, which can mitigate the influence of the malicious agents and at the same time reduce the communication. Each regular agent updates its state based on a given rule using its neighbors' information. Only when the triggering condition is satisfied, they send their current states to their neighbors. Otherwise, the neighbors will continue to use the state received in the last time. Assuming that a bound on the number of malicious nodes is known, we propose two update rules with event-triggered communication. They follow the so-called mean subsequence reduced (MSR) type algorithms and ignore values received from potentially malicious neighbors. We provide full characterizations for the necessary connectivity in the network for the algorithms to perform correctly, stated in terms of the notion of graph robustness. A numerical example is provided to demonstrate the effectiveness of the approach.

(2) We further extend the event-triggered update scheme for the problem of multi-agent consensus in the presence of faulty and malicious agents within the network. To save memory resources, we focus on the case where the agents take integer (or quantized) values. This quantization approach is moreover combined with the event-based communication protocols for solving the resilient consensus problem. To keep the regular agents from being affected by the behavior of faulty agents, algorithms of the MSR type are employed, where neighbors taking extreme values are ignored in the updates. Different from the real-valued case, the quantized version requires the update rule to be randomized. We characterize the error bound on the achievable level of consensus among the agents as well as the necessary structure for

the network in terms of the notion of robust graphs. We verify via a numerical example the effectiveness of the proposed algorithms.

(3) We study the problem of resilient consensus in multi-agent networks with bounded input constraints. To save energy resources, model predictive control schemes are introduced to solve the resilient consensus problem with input constraints under synchronous and asynchronous communications. Each regular agent solves a constrained finite-time optimal problem with the states of its neighbors and updates its state based on a predetermined update rule. Assuming that the maximum number of malicious nodes is known, we derive algorithms which ignore the large and small values from neighbors to avoid the influence of the malicious nodes. It is guaranteed to attain resilient consensus under the topological condition expressed in terms of graph robustness. Simulation examples are provided to demonstrate the effectiveness of the proposed algorithm.

(4) To save graph resources, several modified MSR algorithms are proposed to solve resilient consensus problem for the case of mobile adversary models. We first discuss the three typical mobile malicious models in the area of computer science and apply them to the resilient consensus problem in multi-agent systems. We check that the related results for binary agreement in complete graphs can guarantee approximate resilient consensus. Moreover, we extend the mobile malicious models to non-complete graphs and propose several novel protocols which are guaranteed to work under certain classes of network connectivity conditions. In addition, based on the so-called Garay's mobile malicious model, we improve the update rules for the cured agents to reduce the necessary connections. Numerical examples are provided

to check the efficacy of our results.

Acknowledgements

Now it is time to summarize my road towards Ph.D in Tokyo Institute of Technology. It is hard to believe that I have spent three years here. In the early period of my Ph.D, I have no knowledge about multi-agent systems or cyber physical security, not to find a direction to begin my studies. Fortunately, my supervisor, Professor Hideaki Ishii, is a very patient and wise man. We kept on discussing about the possible directions and I am really inspired a lot from the discussions. There is no doubt that I cannot complete any chapter of this thesis without his support. Words are not enough to express my gratitude.

I would like to thank Professor Jose M. Maestre, we had many discussions about model predictive control and that helped me to complete the contents of Chapter 5. I am very lucky to have had fruitful discussions with Professor Xavier Défago and Professor François Bonnet. The contributions in Chapter 6 are the results of our discussions. The suggestions given by Professor Défago and Professor Bonnet significantly enhanced the quality of this part. I am very grateful for their generous help.

I would like also to appreciate Professor Isao Ono, Professor Yoshihiro Miyake and Professor Masayuki Yamamura, who kindly served as the examiners of my thesis committee. Thank you for reading my thesis and providing me the positive and constructive feedback.

Last but not least, I'd like to thank my parents for their supporting. To study aboard alone is challenging and their emotional support is the base of my study in the past three years.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In recent years, because of the development of communication networks, new opportunities and challenges have been brought to various industrial and societal domains from electric power grids to smart cities. To build a connection between the traditional computer network and physical components becomes a hot topic. The related problems have attracted a lot of researchers. Cyber-physical systems (CPS) form the mechanism that combines computational elements and sensors or other devices. The fusion of physical, computational and communication elements plays an important role in the CPS. Unlike more traditional embedded systems, CPS is usually a network of interacting elements rather than alone devices ([2]). A typical application of CPS is the sensor based autonomous systems with communications. For example, wireless sensor networks to monitor the environment and share the processed information to a central or neighbor agent. Other applications of CPS include smart grid ([22; 35]), industrial cloud technologies ([13]), smart cities ([91]) and so on.

A focus on the control system aspects of CPS is the Cyber-physical security

problem ([47]). A famous example of cyber attacks affect the CPS is the Stuxnet, which is a malicious computer worm for attacking the modern supervisory control and data acquisition (SCADA) and programmable logic controllers (PLCs) systems. It is believed that Stuxnet cause serious damage to Iran's nuclear program. In 2015, a Malware called BlackEnergy attacks the power grid in Ukraine and leads to a power outage as a result. Because of the wide applications of CPS, the security problems in CPS are becoming a hot topic recently.

It is recognized that cyber security for such systems is a critical issue since the extensive use of networks for the interactions among agents creates numerous vulnerabilities for potential attacks (e.g., [70]). Applications such as those in robotics involve physical aspects, and hence, different from cyber attacks limited to the domain of information technology, attacks may lead to damages in equipments or even accidents. Conventional control approaches usually cannot guarantee control objective in an unreliable network. Any fault or cyber attack by an external attacker can seriously affect the system behavior and make it difficult for control objective to be attained. Novel approaches to enhance the reliability and resiliency has gained much attention in networked control ([38; 70]). The desired goal is to address how to mitigate the influence of uncertainties in the system and to design resilient algorithms to guarantee control objective even under worst-case scenarios.

Another issue of the background is resources saving. The necessity of resource saving has been claimed in a wide range of studies. The resource saving in computer science([4; 89]) and control theory ([26]) mainly focus on the saving of communication resources, energy resources and computation resources. Popular approaches include event-based control, quantization, model predictive control and so on.

This thesis follows the general background of security problems in CPS, and

focus on the consensus type problems in CPS. The goal of consensus problem is to reach a common value or interval within in a safety area. Each agent can only use information from neighbor agents. Applications of consensus problems include clock synchronization ([33; 42; 73]), rendezvous ([51; 63]), formation control ([31; 59]), PageRank ([29; 30]) and so on. In this research, we focus on the issue of security in multi-agent systems. One of the fundamental problem is the so called resilient consensus problem, which is the consensus problem with some adversary agents inside. In such problems, the regular agents are trying to reach consensus in cooperate with each other. The adversary agents can update their values arbitrarily, which may affect the updates of regular agents. Our goal is to make the regular agents to reach consensus in a safety interval and in addition, save resources such as computation resources, energy resources, communication resources.

1.2 Cyber security of multi-agent systems

As mentioned above, the developing of networks and communication technology has made the system control to be more convenient and efficient. However, the cyber security problem also becomes a critical issue. Malicious attacks can lead the systems to undesirable operations or even accidents. Safe distributed algorithms are sufficiently discussed in computer science (e.g. [40; 48; 78]) and control (e.g. [61; 67]).

1.2.1 Overview of cyber attacks

Cyber attacks in multi-agent systems are categorized into Denial-of-Service (DoS) attack, replay attack, zero dynamics attack, false data injection attack and so on ([79]). DoS attack mainly focus on the communication connections in the network,

which leads to the failures in data communication or packet losses ([12; 65]). Replay attack includes recording and replaying. The attacker first record the dynamics of system and then replay the recorded data ([57]). Zero dynamics attack requires the attacker has a perfect system knowledge. The designed zero dynamics attacks can guarantee the residue to be zero so that such attacks cannot be detected by residuals ([79]). False data injection attack, which is trying to modify the agent values in the multi-agent systems and then affects the data integrity ([58]). In this research, we focus on the adversary behaviors in each agent. The false data injection behavior is called malicious, and the DoS behavior is called jamming respectively. We assume that every regular agent knows the maximum number of adversary agent in the whole graph is F, which is called F-total model. We analyze the graph condition under the related resilient consensus algorithms, which is called robust graph.

Another attack model comes from the works of computer science ([5; 9]), which is called mobile malicious model. In such models, the malicious agent is dynamic and it can move at any time step. Based on the behavior of the malicious movement and left infected agent, more detailed mobile models such as Buhrman's model ([9]), Garay's model ([24]), Bonnet's model ([5]) are proposed. Based on each model, the related resilient algorithm is proposed and the graph condition is also different from the static malicious model.

1.2.2 Overview of security solutions

In multi-agent systems, there are mainly two popular techniques to deal with the effect of malicious attacks:

1. **Fault Detection and Isolation (FDI)**

   In these solutions, each regular agent is equipped with a observer to identify the possible malicious agents using the past information. Such solutions

are called the Fault Detection and Isolation problems ([64; 76]). The main purpose of FDI is to develop an algorithm to detect the malicious agents and then avoid the influence of them. However, to detect the malicious behavior requires much information and it is difficult to detect every malicious cases. Moreover, such algorithms usually require the whole topology of graphes, which is difficult in distributed algorithms. In this research, we mainly focus on the resilient control approaches.

2. **Resilient control**

In such approaches, the fault detection ability is not necessary. Each regular agent ignores the furthest values from itself and then the misguide from malicious agent can be mitigated. In the area of distributed algorithms in computer science, resilient versions of consensus algorithms have long been studied (see, e.g., [15; 40; 48]), and our work follows this line of research. For each regular agent, a simple but effective approach to reduce the influence of potentially misleading information is to ignore the agents whose states are the most different from its own. It is assumed that the nodes know a priori the maximum number F of adversarial agents in the network. Hence, it is useful to remove the F largest values as well as the F smallest values among those received from the neighbors. This class of algorithms are sometimes called the mean subsequence reduced (MSR) algorithms and has been employed in computer science (e.g., [52; 82]), control theory (e.g., [16; 45; 92]), and robotics (e.g., [25; 63; 69]). The sketch of MSR algorithm is shown in Fig. 1.1. An important recent progress lies in the characterization of the necessary requirement on the topology of the agent networks. This was initiated by [45; 82], where the relevant notion of robust graphs was proposed. It is also remarked that, as a different class of cyber attacks, the effects of jamming and denial-of-service attacks on multi-agent

consensus have recently been analyzed in [41; 74]. In [77], a resilient version of distributed optimization is studied by employing MSR-like mechanisms to remove outliers in neighbors. A resilient state estimation approach for linear time-invariant systems is discussed in [56] to deal with the fault in the networks.

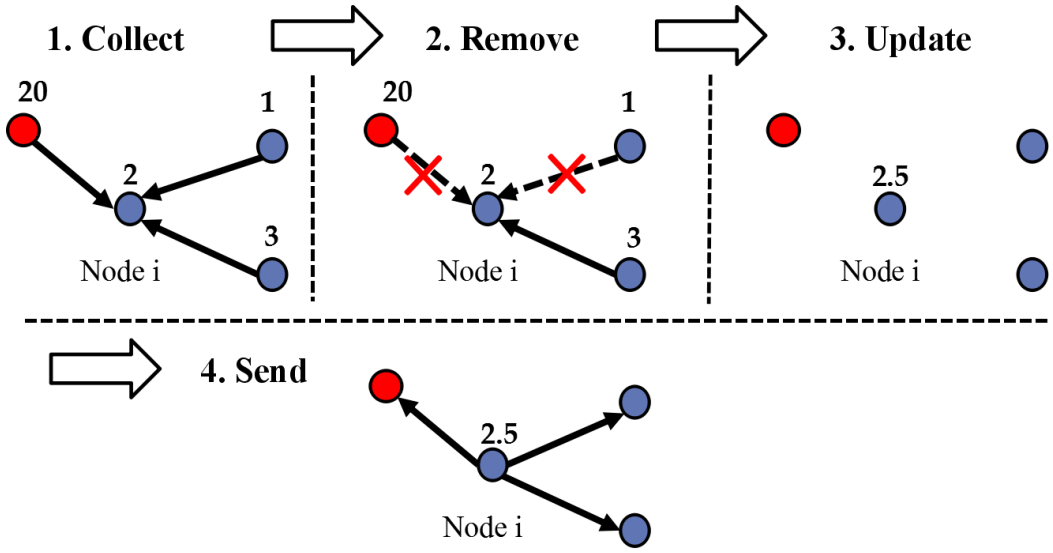


Figure 1.1: Typical MSR algorithm with $F = 1$

1.3 Resilient consensus problems with limited resources

In large-scale multi-agent systems, consensus problems form one of the fundamental problems (e.g., [55]). There, agents interact locally and exchange their information with each other in order to arrive at the global objective of sharing a common value. In an uncertain environment where faults or even adversarial attacks can be present, it is of great importance to defend consensus algorithms by raising their security levels so as to avoid being influenced by such uncertainties

in their decision makings. In this context, adversarial agents are those that do not follow the given algorithms and might even attempt to keep the nonfaulty, regular agents from reaching consensus.

In this research, we emphasis on the resource saving features in consensus problems. Four resources are discussed: Communication resources, memory resources, energy resources and connection resources.

Resilient consensus with limited communication

In Chapter 3, we develop distributed protocols for resilient consensus with a particular emphasis on the communication loads for node interactions. We reduce the transmissions in MSR algorithms through the so-called event-triggered protocols (e.g., [27]). Under this method, nodes make transmissions only when necessary in the sense that their values sufficiently changed since their last transmissions. In certain cases, the agents may make only a finite number of transmissions to neighbors. The advantage is that the communication can be greatly reduced in frequency and may be required only a finite number of times, while the tradeoff is that the achievable level of consensus may be limited, leaving some gaps in the agents' values. Time-triggered protocols may be a simpler way to reduce the communication load, but will not be able to determine when to stop the communication.

More concretely, we develop two protocols for resilient consensus under event-based communication. Their convergence properties are analyzed, and the requirement for the network topology is fully characterized in terms of robust graphs. We will show through a numerical example how the two protocols differ in the amounts of communication needed for achieving consensus. Event-based protocols have been developed for conventional consensus without malicious agents in, e.g., [19; 26; 39; 49; 53; 54; 75]. Related results can be found in [33], where

event-based consensus-type algorithms are developed for the synchronization of clocks possessed by the nodes in wireless sensor networks (WSNs).

The difficulty in applying event-based communication in the context of resilient consensus based on MSR algorithms is due to the handling of the errors between the current values and their last transmitted ones. In our approach, we treat such errors as noise in the system. This approach can be seen as an extension of [42], where a resilient version of the WSN clock synchronization problem in [33] mentioned above is analyzed; the exchange of two clock variables creates decaying noises in the consensus-type algorithms. By contrast, in our problem setting, the errors are due to triggering and do not entirely decay to zero. Moreover, we study a different class of adversarial nodes as we clarify later.

Another feature of Chapter 3 is that we deal with event-driven protocols for consensus algorithms in the discrete-time domain. This is in contrast to the conventional works that deal with event-based consensus in continuous time (e.g., [19; 39; 49; 75]). In such cases, the agents must continuously monitor their states to detect when their states reach the thresholds for triggering events. This mechanism may require special resources for computation. Furthermore, events with short intervals may occur, which can result in undesirable Zeno behaviors. On the other hand, there are works such as [26; 53; 54], where sampled-data controllers are employed for agents with system dynamics in continuous time.

It is interesting to note that in discrete time, event-based consensus algorithms have to be designed differently. This issue has also been discussed in the work [33], which essentially deals with discrete-time asynchronous update rules without adversaries. It is emphasized that in the presence of attacks, this aspect seems even more crucial. In Chapter 3, we present two resilient consensus algorithms, but also discuss a third potential approach. The differences among them are modest: At the updates, each agent has the option of using its own state or its

own last transmitted state. We will however see that analysis methods can differ, leading to various levels of conservatism in the bounds on the parameters for the event triggering functions.

Our work follows the line of research on MSR algorithms. It is the first to introduce event-based communication among the agents. Recently, resilient consensus problems based on MSR have gained much attention, and we would like to discuss some works in the following. The early works [45; 82] dealt with first-order agents with synchronous updates. In [16], MSR-type algorithms are developed for agents having second-order dynamics, which may hence be applicable to autonomous vehicles. Moreover, in [17], delays in communication as well as asynchronous updates are taken into account. The work [44] studied the MSR-based resilient synchronization problem in a more general setting with agents having higher-order dynamics, operating in continuous time. While most studies mentioned so far deal with agents whose states take real values, the work [18] considers agents with quantized (i.e., integer-valued) states. Also, there is a line of graph theoretic studies (e.g., [81; 92; 93]), which discuss methods to identify the robustness of certain classes of graphs with specified levels of robustness, for both undirected and directed graphs.

Resilient consensus with limited memory

In order to save memories in multi-agent systems, we propose a quantized approach into the updates. The focus of Chapter 4 is to develop distributed protocols for resilient consensus problems by taking account of limited capabilities in communications and computations of the agents. To this end, we combine the effects of quantization and event-triggered communication. It turns out that even for the case without adversarial agents, this combination has not been studied much in the literature (see, e.g., [90]).

Quantization in consensus has been addressed in a number of recent works (e.g., [3; 10; 11; 18; 34; 36; 43]). Due to the states taking integer values, the system operates over a discrete state space, and thus the analysis method differs from the real-valued cases. In particular, it is known to be crucial to incorporate randomization in the algorithms (e.g., [80]) to avoid the states to reach steady states with no consensus. This can be done by randomization in the updating times of agents (sometimes called gossiping) or by the use of probabilistic quantizers. In this paper, we take the latter approach and extend the results of [18], which corresponds to simpler case where the agents communicate at every time step.

On the other hand, the updating times are regulated by an event-triggered scheme. It enable us to reduce the amount of transmissions among the agents. The idea is to make new transmissions only when necessary in the sense that the new data is sufficiently different from the previously transmitted one. Such schemes have been employed in various problems in multi-agent consensus [19; 21; 26; 75]. It is interesting to note that most of these works deal with consensus problems in the continuous-time domain; in this case, the triggering condition must be checked continuously and may consume a lot of computation resources. By contrast, our study is carried out in discrete time, which is more suitable for digital implementation; for related results, see also [33].

In Chapter 3, we have studied MSR-type algorithms for the case when agents take real-valued states. Here, we develop parallel results for the quantized case and, in particular, derive necessary and sufficient conditions for achieving resilient consensus. In general, while event-triggered schemes are effective in decreasing the frequency of communication, the achievable level of consensus can be limited, potentially leaving some gaps among the state values of the agents. Our results expose the tradeoff between the amount of communication and the size of the

gap. As in Chapter 3, we provide two update schemes whose difference may appear minor, but results in different upper bounds on the gaps for approximate consensus.

Resilient consensus with limited energy

In Chapter 5, in order to save energy, we consider the resilient consensus problem with input constraint. In addition, we formulate an energy function and study an optimization-based consensus problem, where the agents are subject to input constraints due to limitations in the actuators. The agents aim at reaching the global objective of finding a common value through their interactions. To this end, we employ model predictive control techniques. At each update time, the nonfaulty, regular agents individually solve finite-horizon optimal control problems. They then implement all or some of the optimal control inputs calculated. By repeating the process, they are guaranteed to come to agreement. Such techniques have been implemented, for example, in platoon control of vehicles [1], [94].

We briefly discuss the background on model predictive control (MPC) in the context of multi-agent systems. Distributed MPC-based schemes have been studied for cooperative control of agents with general dynamics in, e.g., [20], [37]. Furthermore, to deal with uncertainties within agent systems, robust distributed MPC methods are developed for linear systems [68] and nonlinear systems [46]. It is however noted that all results mentioned above consider stabilization problems for a priori known setpoints while they use the cost functions as Lyapunov functions.

Even though it is desirable to achieve optimal consensus by distributed MPC scheme, there have been few results for the agent states to agree upon a point not specified beforehand. The work [32] employs negotiation techniques to reach op-

timal consensus by implementing the primal dual decomposition and incremental sub-gradient algorithm. In [23], consensus problems are studied for agents having first-order and second-order dynamics, and conditions for achieving consensus are developed by exploiting some geometry properties of the optimal path. We follow the analysis approach of [23] and apply it to our problem formulation. The work [60] provides a framework for the discrete-time case of distributed model predictive control. Regarding security issues, in [95], replay attacks on formation control of vehicle networks are studied from the MPC perspective. In [83], a resilient distributed MPC-type algorithm is proposed and shown to be effective via simulation studies.

Resilient consensus with limited connection

In Chapter 6, we discuss resilient consensus problem in the mobile malicious model. Compared with the conventional computer science works that discuss the mobile malicious model in complete graphs ([9; 24; 72]), we concentrate on the non-complete graph and robust graph. It is obvious that our protocols require less connections. The mobile malicious behaviors have closer relationship with the dynamic multi-agent systems, for example, mobile sensor networks ([62]), mobile robot networks ([14; 66]), epidemic models ([8]), mobile ad hoc networks ([50]). Protecting mobile agents against malicious hosts is another a popular topic ([28; 71]).

It is interesting to remind that most mobile malicious behaviors are discussed under the complete graph. There is limited works discussing the mobile malicious behaviors under the non-complete graphs. Pierpaoli et al discussed the fault tolerant control approaches for networked mobile robots under a non-complete graph in [66] and follows the FDI line to propose a two-stage technique for solving the FDI problem. In Chapter 6, we follow the resilient control line and propose

several novel protocols to solve the resilient consensus problem under the non-complete graphs.

1.4 Contributions of the thesis

This thesis focuses on the topic of resilient consensus problems in multi-agent systems. In such problems, regular agents are trying to reach agreement on a safe value or interval by local communication. Meanwhile, the adversary agents are allowed to know the global information and trying to mislead the regular agents. Such problems has been long studied in the area of distributed algorithms in computer science since 1980s. However, from the viewpoint of multi-agent control such as unmanned aerial vehicle (UAV) and wireless sensor networks, new motivations and problem settings are recently recognized. Recently, from the multi-agent control viewpoint, several works such as [17; 18; 45] are discussing the convergence of different types of resilient consensus problems. From synchronous systems to asynchronous systems, first-order systems to second-order systems, fruitful contributions are found in this background recently. This thesis also follows the research line. Compared with the previous works, not only the convergence of new resilient consensus algorithms, we also pay attention to the resource saving features of such algorithms.

Based on the type of adversary agents, we would like to explain the contributions of the thesis in a more explicit way.

Static adversary model

The first three major topics of this thesis are based on the static adversary model, which is widely applied in the multi-agent control works. We can highlight the contributions of these parts in three aspects as follows:

1. **New protocols for resilient consensus with advantage of communication saving**

   This topic is motivated by the performance of conventional MSR algorithms. In the time based MSR algorithms, the communication is happening at each time. Even the regular agents have reached resilient consensus, the communication is still happening. Our goal is to stop communicating when the regular agents reach resilient consensus. Motivated by the communication reduction performance of event-based protocols in continuous systems, we are trying to apply similar protocols to resilient consensus problems and analyze the convergence of such algorithms. We provide full characterizations for the necessary connectivity in the network for the algorithms to perform correctly, which are stated in terms of the notion of graph robustness.

2. **New protocols for resilient consensus with advantage of memory saving**

   This study follows the basic framework of event-based protocols for resilient consensus. We extend quantized versions of event-based MSR in Chapter 4. Compared with the real-valued version, these protocols have the advantage of memory saving since all regular states are integer. In addition, the quantized version requires the update rule to be randomized. We characterize the error bound on the achievable level of consensus among the agents as well as the necessary structure for the network in terms of the notion of robust graphs.

3. **New protocols for resilient consensus with advantage of energy saving**

   This study is motivated by applying resilient control to autonomous vehicles and robots. In many applications, there exists an upper bound for control

inputs. Moreover, the energy consumption is also a serious problem. Our study emphasis on such problems and we formulate the resilient consensus problem with input constraint in Chapter 5. Each regular agent solves a constrained finite-time optimal problem with the states of its neighbors and updates its state based on a predetermined update rule. Schemes are proposed to solve the problem with synchronous and asynchronous communications.

Mobile adversary model

Another topic of this thesis is based on the mobile adversary model, which is mainly discussed in distributed algorithms in computer science. From the viewpoint of multi-agent control, limited literature could be found. In this research, we are trying to apply the related mobile adversary models into the resilient consensus problem. The main features of this part are two aspects: (i). Based on the three typical mobile malicious models in computer science works ([5; 9; 24]), we apply them to the resilient consensus problems both complete and non-complete graphs. (ii). Based on Garay's mobile malicious model in [24], we have improved the update rules for the cured agents and reduced the necessary connections as the result.

Table 1.1: Contributions of this thesis

	Time-triggered	Event-triggered
Synchronous	[45]	This work
Asynchronous	[17]	This work
Quantized	[18]	This work
Input constraint	This work	Open problem
Mobile malicious	This work	Open problem

1.5 Outline of the thesis

This thesis is organized as follows.

In Chapter 2, we introduce some general notions from graph theory to robust graphes. Then, the adversary models and notion for resiliency are formulated.

In Chapter 3, we focus on the resilient consensus problem with communication resources saving. We develop event-triggered update rules which can mitigate the influence of the malicious agents and at the same time reduce the communication.

Chapter 4 discusses the saving of memory resources in resilient consensus problems. we focus on the case where the agents take integer (or quantized) values. Different from the real-valued case, the quantized version requires the update rule to be randomized.

Chapter 5 studies the problem of resilient consensus in multi-agent networks with bounded input constraints. Each regular agent solves a constrained finite-time optimal problem with the states of its neighbors and updates its state based on a predetermined update rule.

Chapter 6 focuses on the resilient consensus problem under mobile malicious models. Three typical mobile malicious models are applied to several non-complete graphs. Three novel protocols are proposed to solve the resilient consensus problem with related mobile models.

Finally, Chapter 7 gives a summary for the results and open problems. Some interesting directions for the future research are also given in this chapter.

Chapter 2

Preliminaries

In this chapter, as the basis of the thesis, some general notions from graph theory to robust graphes are introduced at first. Then, we formulate the adversary models and give the notion for resiliency.

2.1 Multi-agent networks on graphs

Some basic notations related to graphs are introduced for the analysis in this thesis.

Consider the directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting of n nodes. Here the set of nodes is denoted by $\mathcal{V} = \{1, 2, \ldots, n\}$ and the edge set by $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The edge $(j, i) \in \mathcal{E}$ indicates that node j can send a message to node i and is called an incoming edge of node i. Let $\mathcal{N}_i = \{j : (j, i) \in \mathcal{E}\}$ be the set of neighbors of node i. The number of neighbors of node i is called its degree and is denoted as $d_i = |\mathcal{N}_i|$ The path from node i_1 to node i_p is denoted as the sequence $(i_1, i_2, \ldots, i_p)$, where $(i_j, i_{j+1}) \in \mathcal{E}$ for $j = 1, 2, \ldots, p-1$. The graph $\mathcal{G}$ is said to have a spanning tree if there exists a node from which there is a path to all other nodes of this graph.

2.2 Robust graphs

To establish resilient consensus results, an important topological notion is that of robustness of graphs [45].

Definition 2.2.1. The graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is called (r, s)-*robust* $(r, s < n)$ if for any two nonempty disjoint subsets $\mathcal{V}_1, \mathcal{V}_2 \subseteq \mathcal{V}$, one of the following conditions is satisfied:

1. $\mathcal{X}^r_{\mathcal{V}_1} = \mathcal{V}_1$,
2. $\mathcal{X}^r_{\mathcal{V}_2} = \mathcal{V}_2$,
3. $|\mathcal{X}^r_{\mathcal{V}_1}| + |\mathcal{X}^r_{\mathcal{V}_2}| \geq s$,

where $\mathcal{X}^r_{\mathcal{V}_i}$ is the set of all nodes in $\mathcal{V}_i$ which have at least r neighbors outside $\mathcal{V}_i$ for $i = 1, 2$. The graph is said to be r-robust if it is $(r, 1)$-robust.

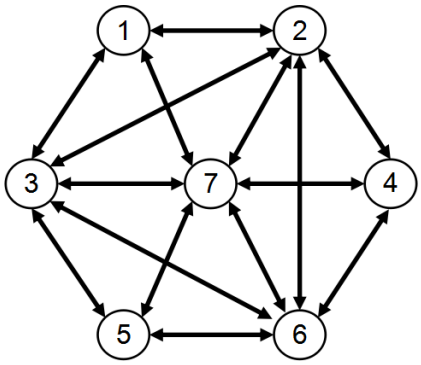


Figure 2.1: Network topology with $(3, 3)$-robustness

In Fig. 2.1, we display an example graph with seven nodes. It can be checked to have just enough connectivity to be (3,3)-robust. If any of the edges are removed, this level of robustness will be lost.

We summarize some basic properties of robust graphs [45]. Here, the ceil function $\lceil y \rceil$ gives the smallest integer greater than or equal to y.

Lemma 2.2.1. An (r, s)-robust graph $\mathcal{G}$ satisfies the following:

1. $\mathcal{G}$ is (r', s')-robust, where $0 \leq r' \leq r$, $1 \leq s' \leq s$, and in particular, it is r-robust.

2. $\mathcal{G}$ has a directed spanning tree. Moreover, it is 1-robust if and only if it has a directed spanning tree.

3. $r \leq \lceil n/2 \rceil$. Furthermore, it holds $r = \lceil n/2 \rceil$ if and only if $\mathcal{G}$ is a complete graph.

4. The degree d_i for $i \in \mathcal{V}$ is lower bounded as $d_i \geq r + s - 1$ if $s < r$ and $d_i \geq 2r - 2$ if $s \geq r$.

Moreover, a graph $\mathcal{G}$ is (r, s)-robust if it is $(r + s - 1)$-robust.

In consensus problems, the property 2) in the lemma is of interest. Robust graphs may not be strongly connected in general, but this property indicates that the notion of robust graphs is a generalization of graphs containing directed spanning trees, which are of great relevance in the literature of consensus [55].

As we will see, robust graphs play a key role in characterizing the necessary network structure for achieving resilient consensus. It should however be noted that checking the robustness of a given graph involves combinatorial computation and is thus difficult in general [81; 92; 93].

2.3 Adversary model and resiliency notions

For each regular node in the set $\mathcal{R}$, its state $x_i(k)$ is updated by

$$x_i(k+1) = x_i(k) + u_i(k). \tag{2.1}$$

where $u_i(k)$ is the control given by

$$u_i(k) = \sum_{j \in \mathcal{N}_i} a_{ij}(k) \left(x_j(k) - x_i(k) \right). \tag{2.2}$$

For each adversarial node i in the set $\mathcal{A}$, its control $u_i(k)$ can take arbitrary values at any k. Such nodes may have knowledge on the entire network including its topology, the values of all normal nodes, and their update rules. In this respect, we take account of their worst-case behaviors. For their communication, we employ the malicious model introduced in [45] and the jamming model as follows:

Definition 2.3.1. *Two adversarial classes are given as follows:*

- Malicious*: We say that an adversarial agent is malicious if it makes updates in its value arbitrarily at each time and sends the same value to all of its neighbors each time a transmission is made.*
- Jamming*: We say that an adversarial agent is jamming if it does not send any value to any of its neighbors each time a transmission is made.*

Adversarial nodes more difficult to deal with are those that can transmit different values to different neighbors in an arbitrary way. Such nodes are referred to as being Byzantine [82]. The motivation for considering malicious nodes as defined above comes, for example, from the applications of WSNs, where sensor nodes communicate to their neighbors by broadcasting their data.

We also set a bound on the number of malicious nodes in the network. In this thesis, we will deal with networks of the so-called F-total model as defined below.

Definition 2.3.2. (*F-total model*): For $F \in \mathbb{N}$, we say that the adversarial set $\mathcal{A}$ follows an *F-total* model if $|\mathcal{A}| \leq F$.

Let the number of malicious agents be denoted by $N_m = |\mathcal{A}|$. Then, let $|\mathcal{R}| = |\mathcal{V}| - N_m$ be the number of regular agents.

Now, we introduce the notion of resilient consensus for multi-agent systems.

Definition 2.3.3. (*Resilient consensus*): Given $c \geq 0$, if for any possible sets and behaviors of the malicious agents and any initial state values of the regular nodes, the following conditions are satisfied, then the multi-agent system is said to reach resilient consensus at the error level c:

1. Safety condition: There exists an interval $\mathcal{S} \subset \mathbb{R}$ such that $x_i(k) \in \mathcal{S}$ for all $i \in \mathcal{R}$, $k \in \mathbb{Z}_+$.

2. Consensus condition: For all $i, j \in \mathcal{R}$, it holds that $\limsup_{k\to\infty} |x_i(k) - x_j(k)| \leq c$.

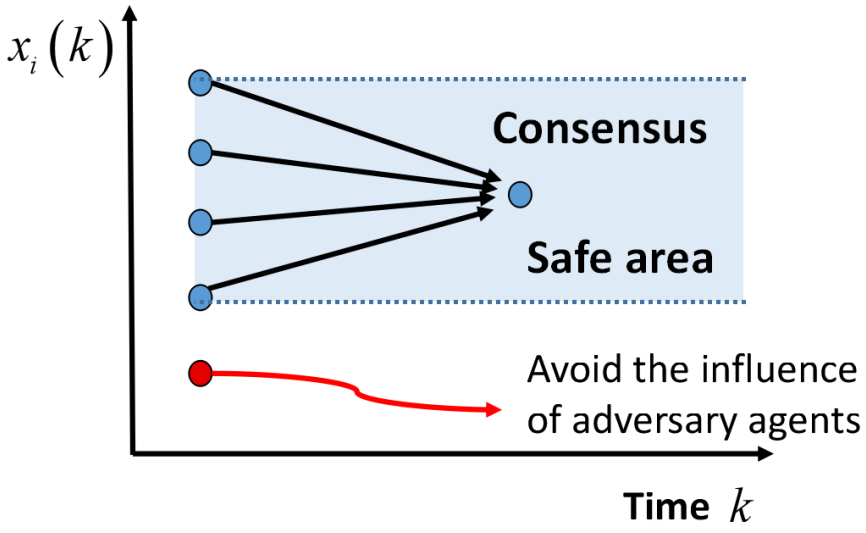


Figure 2.2: Resilient consensus problem

Chapter 3

Resilient Consensus Through Event-based Communication

In this chapter, we consider resilient versions of discrete-time multi-agent consensus in the presence of faulty or even malicious agents in the network. In particular, we develop event-triggered update rules which can mitigate the influence of the malicious agents and at the same time reduce the communication. Each regular agent updates its state based on a given rule using its neighbors' information. Only when the triggering condition is satisfied, the regular agents send their current states to their neighbors. Otherwise, the neighbors will continue to use the state received the last time. Assuming that a bound on the number of malicious nodes is known, we propose two update rules with event-triggered communication. They follow the so-called mean subsequence reduced (MSR) type algorithms and ignore values received from potentially malicious neighbors. We provide full characterizations for the necessary connectivity in the network for the algorithms to perform correctly, which are stated in terms of the notion of graph robustness. A numerical example is provided to demonstrate the effectiveness of the proposed approach. This part is published in [87].

3.1 Problem formulation

We introduce the event-based protocol for the regular nodes to achieve consensus. It can be outlined as follows: At each discrete-time instant $k \in \mathbb{Z}_+$, the nodes make updates, but whether they transmit their current values to neighbors depends on the triggering function. More concretely, each node i has an auxiliary variable which is its state value communicated the last time and compares it with its own current state. If the current state has changed sufficiently, then it will be sent to its neighbors and the auxiliary variable will be replaced.

The update rule for agent i is described by

$$x_i(k+1) = x_i(k) + u_i(k), \tag{3.1}$$

where $x_i(k) \in \mathbb{R}$ is the state and $u_i(k)$ is the control given by

$$u_i(k) = \sum_{j \in \mathcal{N}_i} a_{ij}(k) \left(\hat{x}_j(k) - x_i(k)\right). \tag{3.2}$$

Here, $\hat{x}_j(k) \in \mathbb{R}$ is an auxiliary state, representing the last communicated state of node j at time k. It is defined as

$$\hat{x}_j(k) = x_j(t_l^j), \quad k \in [t_l^j, t_{l+1}^j),$$

where $t_0^j, t_1^j, \ldots$ denote the transmission times of node j determined by the triggering function to be given below. The initial values $x_i(0)$, $\hat{x}_j(0)$ are given, and $a_{ij}(k)$ is the weight for the edge (j, i). Also, let $a_{ii}(k) = 1 - \sum_{j \in \mathcal{N}_i} a_{ij}(k)$. Assume that $\gamma \leq a_{ij}(k) < 1$ when $a_{ij}(k) \neq 0$ for $i, j \in \mathcal{V}$, where γ is the lower bound with $0 < \gamma \leq 1/2$. In the resilient consensus algorithms to be introduced, the neighbors whose values are used for updates change over time, and hence, the weights

$a_{ij}(k)$ are time varying. The update rule above can be seen as a discrete-time counterpart of the event-based consensus algorithms in, e.g., [26; 49; 75].

We now introduce the triggering function. Denote the error at time k between the updated state $x_i(k+1)$ and the auxiliary state $\hat{x}_i(k)$ by $e_i(k) = \hat{x}_i(k) - x_i(k+1)$ for $k \geq 0$. Then, let

$$f_i(k) = |e_i(k)| - \left(c_0 + c_1 \mathrm{e}^{-\alpha k}\right), \tag{3.3}$$

where c_0, c_1, and $\alpha > 0$ are positive constants. If $f_i(k) > 0$, agent i transmits its new state $x_i(k+1)$ to the neighbors at time k. This mechanism will be discussed further later.

3.2 Protocol 1 for event-based consensus

In this section, we outline a distributed protocol to solve the resilient consensus problem. As discussed above, every node makes an update at every time step in a synchronous manner, but only when an event happens, the auxiliary values will be updated and then sent to neighbors. The basis of the algorithm follows those in the works of, e.g., [16; 17; 45]. The algorithm in this chapter is called the event-based mean subsequence reduced (E-MSR) algorithm.

The E-MSR algorithm has four steps as follows:

1. *(Collecting neighbors' information)* At each time step k, every regular node $i \in \mathcal{R}$ uses the values $\hat{x}_j(k), j \in \mathcal{N}_i$, most recently communicated from the neighbors as well as its own value $x_i(k)$ and sorts them from the largest to the smallest.

2. *(Deleting suspicious values)* Comparing with $x_i(k)$, node i removes the F largest and F smallest values from its neighbors. If the number of values larger or smaller than $x_i(k)$ is less than F, then all of them are removed.

The removed data is considered as suspicious and will not be used in the update. The set of the node indices of the remaining values is written as $\mathcal{M}_i(k) \subset \mathcal{N}_i$.

3. *(Local update)* Node i updates its state by

$$x_i(k+1) = x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) \left(\hat{x}_j(k) - x_i(k)\right). \tag{3.4}$$

4. *(Communication update)* Node i checks if its own triggering function $f_i(k)$ in (3.3) is positive or not. Then, it sets $\hat{x}_i(k+1)$ as

$$\hat{x}_i(k+1) = \begin{cases} x_i(k+1) & \text{if } f_i(k) > 0, \\ \hat{x}_i(k) & \text{otherwise.} \end{cases} \tag{3.5}$$

The communication rule in this algorithm shows that only when the current value has varied enough to exceed a threshold, then the auxiliary variable will be updated, and only at this time the node sends its value to its neighbors. This event triggering scheme can significantly reduce the communication burden as we will see in the numerical example in Section 3.4.

The first protocol of this chapter is the E-MSR algorithm as stated above, which will be referred to as Protocol 1. We are now ready to present our main result for this protocol.

We introduce two kinds of minima and maxima of the states of the regular agents: The first involves only the states as $\overline{x}(k) = \max_{i \in \mathcal{R}} x_i(k)$ and $\underline{x}(k) = \min_{i \in \mathcal{R}} x_i(k)$ while the second uses also the auxiliary variables as $\underline{\hat{x}}(k) = \min_{i \in \mathcal{R}}\{x_i(k), \hat{x}_i(k)\}$ and $\overline{\hat{x}}(k) = \max_{i \in \mathcal{R}}\{x_i(k), \hat{x}_i(k)\}$. The safety interval $\mathcal{S}$ is chosen as $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$. It is noted that at initial time, $\hat{x}_i(0)$ need not be the same as $x_i(0)$.

Theorem 3.2.1. Under the F-total model, the regular agents with E-MSR using (3.4) and (3.5) reach resilient consensus at an error level c if and only if the underlying graph is $(F+1, F+1)$-robust. The safety interval is given by $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$, and the consensus error level c is achieved if the parameter c_0 in the triggering function (3.3) satisfies

$$c_0 \leq \frac{\gamma^{|\mathcal{R}|} c}{4|\mathcal{R}|}. \tag{3.6}$$

Proof. (Necessity) This essentially follows from [45], which considers the special case without the triggering function, that is, $c_0 = c_1 = 0$.

(Sufficiency) We first show that the interval $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$ satisfies the safety condition by induction. Note that the update rule (3.4) can be rewritten as

$$x_i(k+1) = a_{ii}(k)x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\hat{x}_j(k), \tag{3.7}$$

where $a_{ii}(k) = 1 - \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)$.

At time $k = 0$, it is clear by definition that $x_i(0), \hat{x}_i(0) \in \mathcal{S}$, $i \in \mathcal{R}$. Suppose that for each regular agent i, $x_i(k), \hat{x}_i(k) \in \mathcal{S}$. Then, for agent i, its neighbors in $\mathcal{M}_i(k)$ take values only in $\mathcal{S}$, since there are agents with values outside $\mathcal{S}$ at most F, and they are ignored in step 2 of the E-MSR. From (3.7), we have $x_i(k+1) \in \mathcal{S}$. Moreover, by (3.5), it follows that $\hat{x}_i(k+1) \in \mathcal{S}$. Thus, $\mathcal{S}$ is the safety interval.

We next establish the consensus condition. Note that for time $k \in (t_l^i, t_{l+1}^i)$ between two triggering instants, we have $f_i(k) \leq 0$. Moreover, for the neighbor node $j \in \mathcal{N}_i$, if $f_j(k) > 0$, then we have $\hat{x}_j(k+1) = x_j(k+1)$. If $f_j(k) \leq 0$, then $\hat{x}_j(k+1) = \hat{x}_j(k) = x_j(k+1) + e_j(k)$. As a result, it holds $\hat{x}_j(k) = x_j(k) + \hat{e}_j(k-1)$

for $k \geq 1$, where

$$\hat{e}_j(k) = \begin{cases} e_j(k) & \text{if } f_j(k) \leq 0, \\ 0 & \text{otherwise.} \end{cases}$$

Note that

$$|\hat{e}_j(k)| \leq c_0 + c_1 \mathrm{e}^{-\alpha k}, \quad \forall k \geq 0. \tag{3.8}$$

Then, we can write (3.7) as

$$x_i(k+1) = a_{ii}(k)x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) \left(x_j(k) + \hat{e}_j(k-1) \right). \tag{3.9}$$

This can be bounded by using the maximum state $\overline{x}(k)$ as

$$\begin{aligned} x_i(k+1) &\leq a_{ii}(k)\overline{x}(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) \left(\overline{x}(k) + \hat{e}_j(k-1) \right) \\ &= \overline{x}(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k-1) \\ &\leq \overline{x}(k) + \max_{j \in \mathcal{M}_i(k)} \left| \hat{e}_j(k-1) \right|. \end{aligned} \tag{3.10}$$

Thus, by (3.8) it follows

$$x_i(k+1) \leq \overline{x}(k) + c_0 + c_1 \mathrm{e}^{-\alpha(k-1)}.$$

Let $V(k) = \overline{x}(k) - \underline{x}(k)$. Then we introduce two sequences given by

$$\overline{x}_0(k+1) = \overline{x}_0(k) + c_0 + c_1 \mathrm{e}^{-\alpha(k-1)}, \tag{3.11}$$

$$\underline{x}_0(k+1) = \underline{x}_0(k) - c_0 - c_1 \mathrm{e}^{-\alpha(k-1)}, \tag{3.12}$$

where $\overline{x}_0(0) = \overline{x}(0) - \sigma_0$, and $\underline{x}_0(0) = \underline{x}(0) + \sigma_0$ with $\sigma_0 = \sigma V(0)$. We next

introduce another sequence $\varepsilon_0(k)$ defined by

$$\varepsilon_0(k+1) = \gamma\varepsilon_0(k) - (1-\gamma)\sigma_0, \tag{3.13}$$

where $\varepsilon_0(0) = \varepsilon V(0)$. Take the parameters ε and σ so that

$$\varepsilon + \sigma = \frac{1}{2}, \quad 0 < \sigma < \frac{\gamma^N}{1-\gamma^N}\varepsilon. \tag{3.14}$$

For the sequence $\varepsilon_0(k)$, let

$$\begin{aligned}
\overline{\mathcal{X}}_0(k, \varepsilon_0(k)) &= \{j \in \mathcal{V} : \ x_j(k) > \overline{x}_0(k) - \varepsilon_0(k)\}, \\
\underline{\mathcal{X}}_0(k, \varepsilon_0(k)) &= \{j \in \mathcal{V} : \ x_j(k) < \underline{x}_0(k) + \varepsilon_0(k)\}.
\end{aligned}$$

These two sets are both nonempty at time $k = 0$ and, in particular, each contains at least one regular node; this is because by definition, $\overline{x}(0) > \overline{x}_0(0) - \varepsilon_0(0)$ and $\underline{x}(0) < \underline{x}_0(0) + \varepsilon_0(0)$.

In the following, we show that $\overline{\mathcal{X}}_0(k, \varepsilon_0(k))$ and $\underline{\mathcal{X}}_0(k, \varepsilon_0(k))$ are disjoint sets. To this end, we must show

$$\overline{x}_0(k) - \varepsilon_0(k) \geq \underline{x}_0(k) + \varepsilon_0(k).$$

By (3.11) and (3.12) for $\overline{x}_0(k)$ and $\underline{x}_0(k)$, we have

$$\begin{aligned}
(\overline{x}_0(k) - \varepsilon_0(k)) - (\underline{x}_0(k) + \varepsilon_0(k)) = &\left(\overline{x}_0(0) + c_0 k + c_1 \frac{1 - \mathrm{e}^{-\alpha(k-1)}}{1 - \mathrm{e}^{-\alpha}}\right) \\
&- \left(\underline{x}_0(0) - c_0 k - c_1 \frac{1 - \mathrm{e}^{-\alpha(k-1)}}{1 - \mathrm{e}^{-\alpha}}\right) - 2\varepsilon_0(k).
\end{aligned} \tag{3.15}$$

Then by substituting $\overline{x}_0(0) = \overline{x}(0) - \sigma_0$ and $\underline{x}_0(0) = \underline{x}(0) + \sigma_0$ into the right-hand

side of (3.15), we obtain

$$\begin{aligned}(\overline{x}_0(k) - \varepsilon_0(k)) - (\underline{x}_0(k) + \varepsilon_0(k)) \\ &= (\overline{x}(0) - \underline{x}(0)) - 2\sigma_0 + 2c_0k + 2c_1\frac{1-\mathrm{e}^{-\alpha(k-1)}}{1-\mathrm{e}^{-\alpha}} - 2\varepsilon_0(k) \\ &= V(0) - 2\sigma V(0) + 2c_0k + 2c_1\frac{1-\mathrm{e}^{-\alpha(k-1)}}{1-\mathrm{e}^{-\alpha}} - 2\varepsilon_0(k). \quad (3.16)\end{aligned}$$

By (3.13) and $0 < \gamma \leq 1/2$, we easily have that $\varepsilon_0(k+1) < \varepsilon_0(k)$, and hence $\varepsilon_0(k) < \varepsilon_0(0) = \varepsilon V(0)$. We thus obtain

$$\begin{aligned}(\overline{x}_0(k) - \varepsilon_0(k)) - (\underline{x}_0(k) + \varepsilon_0(k)) \\ &> (1 - 2\sigma - 2\varepsilon)V(0) + 2c_0k + 2c_1\frac{1-\mathrm{e}^{-\alpha(k-1)}}{1-\mathrm{e}^{-\alpha}} > 0,\end{aligned}$$

where the last inequality holds since $\sigma + \varepsilon = 1/2$ from (3.14). Consequently, it follows that $\overline{\mathcal{X}}_0(k, \varepsilon_0(k))$ and $\underline{\mathcal{X}}_0(k, \varepsilon_0(k))$ are disjoint sets.

From the above, we have that the two sets $\overline{\mathcal{X}}_0(0, \varepsilon_0(0))$ and $\underline{\mathcal{X}}_0(0, \varepsilon_0(0))$ are nonempty with at least one regular node in each and moreover disjoint. Therefore, by the assumption of $(F+1, F+1)$-robustness, there are three cases:

1. All nodes in $\overline{\mathcal{X}}_0(0, \varepsilon_0(0))$ have $F+1$ neighbors or more from outside.
2. All nodes in $\underline{\mathcal{X}}_0(0, \varepsilon_0(0))$ have $F+1$ neighbors or more from outside.
3. The total number of nodes in $\overline{\mathcal{X}}_0(0, \varepsilon_0(0))$ and $\underline{\mathcal{X}}_0(0, \varepsilon_0(0))$ having $F+1$ neighbors or more from outside of its own set is no smaller than $F+1$.

Notice that in any of the three cases, there exists at least one regular agent $i \in \mathcal{R}$ in either $\overline{\mathcal{X}}_0(0, \varepsilon_0(0))$ or $\underline{\mathcal{X}}_0(0, \varepsilon_0(0))$ that has $F+1$ neighbors or more from outside of its own set. In the following, we suppose that this node i belongs to $\overline{\mathcal{X}}_0(0, \varepsilon_0(0))$. A similar argument holds for the case when it is in $\underline{\mathcal{X}}_0(0, \varepsilon_0(0))$.

Now, we go back to (3.9) and rewrite it by partitioning the neighbor node set $\mathcal{M}_i(k)$ of node i into two parts: The nodes which belong to $\overline{\mathcal{X}}_0(k, \varepsilon_0(k))$ and those that do not. Since node i has at least $F+1$ neighbors outside $\overline{\mathcal{X}}_0(k, \varepsilon_0(k))$, the latter set is nonempty. Hence, we obtain

$$\begin{aligned} x_i(k+1) = a_{ii}(k)x_i(k) + \sum_{j \in \mathcal{M}_i(k) \cap \overline{\mathcal{X}}_0} a_{ij}(k)x_j(k) \\ + \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}_0} a_{ij}(k)x_j(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k-1), \end{aligned}$$

where we use the shorthand notation $\overline{\mathcal{X}}_0$ for $\overline{\mathcal{X}}_0(k, \varepsilon_0(k))$. Then, we can bound this from above as

$$\begin{aligned} x_i(k+1) &\leq a_{ii}(k)\overline{x}(k) + \sum_{j \in \mathcal{M}_i(k) \cap \overline{\mathcal{X}}_0} a_{ij}(k)\overline{x}(k) + \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}_0} a_{ij}(k) \left(\overline{x}_0(k) - \varepsilon_0(k)\right) \\ &\quad + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k-1) \\ &= \left(1 - \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}_0} a_{ij}(k)\right) \overline{x}(k) + \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}_0} a_{ij}(k) \left(\overline{x}_0(k) - \varepsilon_0(k)\right) \\ &\quad + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k-1). \end{aligned} \tag{3.17}$$

We next show that $\overline{x}(k) \leq \overline{x}_0(k) + \sigma_0$ (and similarly, $\underline{x}(k) \geq \underline{x}_0(k) - \sigma_0$) by induction. For $k = 0$, by definition, we have $\overline{x}(0) = \overline{x}_0(0) + \sigma_0$. Suppose that $\overline{x}(k) \leq \overline{x}_0(k) + \sigma_0$. Then, from (3.10) and (3.11), we have

$$\begin{aligned} \overline{x}(k+1) &\leq \overline{x}(k) + \max_j |\hat{e}_j(k-1)| \leq \overline{x}(k) + c_0 + c_1 \mathrm{e}^{-\alpha(k-1)} \\ &\leq \overline{x}_0(k) + \sigma_0 + c_0 + c_1 \mathrm{e}^{-\alpha(k-1)} = \overline{x}_0(k+1) + \sigma_0. \end{aligned}$$

Then, (3.17) can be further bounded as

$$\begin{aligned} x_i(k+1) &\leq \left(1 - \sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathcal{X}}_0} a_{ij}(k)\right)(\overline{x}_0(k)+\sigma_0) + \sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathcal{X}}_0} a_{ij}(k)\left(\overline{x}_0(k)-\varepsilon_0(k)\right) \\ &\quad + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k-1) \\ &\leq \overline{x}_0(k) + \left(1 - \sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathcal{X}}_0} a_{ij}(k)\right)\sigma_0 - \sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathcal{X}}_0} a_{ij}(k)\varepsilon_0(k) \\ &\quad + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)|\hat{e}_j(k-1)|. \end{aligned} \tag{3.18}$$

We also show that $\varepsilon_0(k) > 0$ holds for $k = 0, 1, \ldots, |\mathcal{R}|$. It is clear from (3.13) that $\varepsilon_0(k+1) < \varepsilon_0(k)$. Thus we only need to guarantee $\varepsilon_0(|\mathcal{R}|) > 0$. By (3.13), $\varepsilon_0(|\mathcal{R}|)$ can be written as

$$\begin{aligned} \varepsilon_0(|\mathcal{R}|) &= \gamma^{|\mathcal{R}|}\varepsilon_0(0) - \sum_{i=0}^{|\mathcal{R}|-1} \gamma^i(1-\gamma)\sigma_0 \\ &= \gamma^{|\mathcal{R}|}\varepsilon V(0) - \frac{1-\gamma^{|\mathcal{R}|}}{1-\gamma}(1-\gamma)\sigma V(0) \\ &= \left(\gamma^{|\mathcal{R}|}\varepsilon - (1-\gamma^{|\mathcal{R}|})\sigma\right)V(0). \end{aligned}$$

This is positive because we have chosen σ as in (3.14).

Hence, (3.18) can be written as

$$\begin{aligned} x_i(k+1) &\leq \overline{x}_0(k) + (1-\gamma)\,\sigma_0 - \gamma\varepsilon_0(k) + c_0 + c_1 \mathrm{e}^{-\alpha(k-1)} \\ &= \overline{x}_0(k+1) - \varepsilon_0(k+1), \end{aligned} \tag{3.19}$$

where in the inequality, we used the fact that there always exists j not in $\overline{\mathcal{X}}_0(k, \varepsilon_0(k))$. This relation shows that if an update happens at node i, then this node will move out of $\overline{\mathcal{X}}_0(k+1, \varepsilon_0(k+1))$. We note that inequality (3.19)

also holds for the regular nodes that are not inside $\overline{\mathcal{X}}_0(k, \varepsilon_0(k))$ at time k. This means that such nodes cannot move in $\overline{\mathcal{X}}_0(k+1, \varepsilon_0(k+1))$. It is also similar with $\underline{\mathcal{X}}_0(k+1, \varepsilon_0(k+1))$.

Thus, after $|\mathcal{R}|$ time steps, all regular nodes will be out of at least one of the two sets $\overline{\mathcal{X}}_0(|\mathcal{R}|, \varepsilon_0(|\mathcal{R}|))$ and $\underline{\mathcal{X}}_0(|\mathcal{R}|, \varepsilon_0(|\mathcal{R}|))$. We suppose that $\overline{\mathcal{X}}_0(|\mathcal{R}|, \varepsilon_0(|\mathcal{R}|)) \cap \mathcal{R}$ is empty. Then we have $\overline{x}(|\mathcal{R}|) \leq \overline{x}_0(|\mathcal{R}|) - \varepsilon_0(|\mathcal{R}|)$. It hence follows that

$$
\begin{aligned}
V(|\mathcal{R}|) &= \overline{x}(|\mathcal{R}|) - \underline{x}(|\mathcal{R}|) \\
&\leq \overline{x}_0(|\mathcal{R}|) - \varepsilon_0(|\mathcal{R}|) - \underline{x}_0(|\mathcal{R}|) + \sigma_0 \\
&= \overline{x}_0(0) - \underline{x}_0(0) + 2c_0|\mathcal{R}| + 2\sum_{i=0}^{|\mathcal{R}|-1} c_1 \mathrm{e}^{-\alpha i} - \varepsilon_0(|\mathcal{R}|) + \sigma_0 \\
&= (\overline{x}(0) - \sigma_0) - (\underline{x}(0) + \sigma_0) + 2c_0|\mathcal{R}| + 2c_1 \frac{1 - \mathrm{e}^{-\alpha|\mathcal{R}|}}{1 - \mathrm{e}^{-\alpha}} - \varepsilon_0(|\mathcal{R}|) + \sigma_0 \\
&= V(0) + 2c_0|\mathcal{R}| + 2c_1 \frac{1 - \mathrm{e}^{-\alpha|\mathcal{R}|}}{1 - \mathrm{e}^{-\alpha}} - \sigma V(0) - \left(\gamma^{|\mathcal{R}|}\varepsilon - \left(1 - \gamma^{|\mathcal{R}|}\right)\sigma\right) V(0) \\
&= \left(1 - \gamma^{|\mathcal{R}|}(\varepsilon + \sigma)\right) V(0) + 2c_0|\mathcal{R}| + 2c_1 \frac{1 - \mathrm{e}^{-\alpha|\mathcal{R}|}}{1 - \mathrm{e}^{-\alpha}}.
\end{aligned}
$$

By $\varepsilon + \sigma = 1/2$ in (3.14), we have

$$
V(|\mathcal{R}|) \leq \left(1 - \frac{\gamma^{|\mathcal{R}|}}{2}\right) V(0) + 2c_0|\mathcal{R}| + 2c_1 \frac{1 - \mathrm{e}^{-\alpha|\mathcal{R}|}}{1 - \mathrm{e}^{-\alpha}}.
$$

If there are more updates by node i after time $k = |\mathcal{R}|$, this argument can be

extended further as

$$\begin{aligned}
V(l|\mathcal{R}|) &\leq \left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right) V((l-1)|\mathcal{R}|) + 2c_0|\mathcal{R}| + 2c_1\frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}}\mathrm{e}^{-(l-1)\alpha|\mathcal{R}|} \\
&\leq \left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{l} V(0) + \sum_{t=0}^{l-1}\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{l-1-t} \\
&\qquad\qquad \times \left(2c_0|\mathcal{R}| + 2c_1\frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}}\mathrm{e}^{-(t-1)\alpha|\mathcal{R}|}\right) \\
&\leq \left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{l} V(0) + \frac{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{l}}{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)} 2c_0|\mathcal{R}| \\
&\quad + 2c_1\frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}}\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{l}\frac{1-(1-\frac{\gamma^{|\mathcal{R}|}}{2})^{-l}\mathrm{e}^{-\alpha|\mathcal{R}|l}}{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{-l}\mathrm{e}^{-\alpha|\mathcal{R}|}}.
\end{aligned} \tag{3.20}$$

From (3.6), we can easily obtain

$$\limsup_{l\to\infty} V(l|\mathcal{R}|) \leq \frac{2c_0|\mathcal{R}|}{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)} = \frac{4c_0|\mathcal{R}|}{\gamma^{|\mathcal{R}|}} \leq c. \tag{3.21}$$

Now, we show the dynamics of $V(l|\mathcal{R}|+t)$ for $t = 0, 1, \ldots, |\mathcal{R}|-1$. The analysis is similar, and we can obtain an inequality like (3.20), where the only difference is that in the derivation, $V(0)$ is replaced with $V(t)$. From the safety condition, we know that $V(k) \leq |\mathcal{S}|$ for all k. Therefore, we finally arrive at

$$\limsup_{l\to\infty} V(l|\mathcal{R}| + t) \leq \frac{4c_0|\mathcal{R}|}{\gamma^{|\mathcal{R}|}} \leq c.$$

This completes the proof of the consensus condition. □

The above result shows that the multi-agent system is guaranteed to reach resilient consensus despite the presence of F-total malicious agents. First, the width of the safety interval $\mathcal{S}$ is determined by the initial states of the regular

agents. Second, the error that may remain after achieving resilient consensus meets the specified bound c by selecting the key parameter in the triggering function c_0, proportionally to c. This parameter can be set by the designer and, clearly, by taking $c_0 = 0$, exact consensus can be achieved at the expense of having more communications. The role of c_1 and α is to reduce the communication during the transient stage by making the threshold in the triggering function large. We note that the exponential decaying bound by c_1 and α can also decrease the communication in the long run.

As a result of effects of triggering parameters c_0, c_1 and α, c_0 can efficiently reduce the communications and can avoid communications in a long period of time. c_1 and α can efficiently reduce the communications in the initial times and do not affect the consensus error level. However, in the long time effect of reducing communications is not as effective as c_0. We will see the effects of the parameters of the event-triggering function through a numerical example in Section 3.4.

In the literature of event-based consensus, conventional schemes often employ triggering functions whose thresholds go to zero over time, in both continuous- and discrete-time domains (e.g., [19; 26; 49; 53; 54]). By contrast, [39; 75] use thresholds which always take positive values as in our study. In comparison, our upper bound for the consensus error is more conservative. Because of the malicious agents, the analysis cannot apply the methods in previous works and must follow those in resilient consensus problems such as [17]; as a consequence, the bound on consensus errors grows exponentially with $|\mathcal{R}|$ (see (3.21)). In the conventional results of [39; 75], the bounds depend on $|\mathcal{R}|$ linearly as well as on the Laplacian matrix.

A related result for the case of F-local model for the adversarial nodes can be found in [42] with a particular application to clock synchronization in WSNs.

It studies a resilient consensus problem with decaying noise that arises in the system due to the interactions among clock states.

Remark 3.2.1. We should highlight that in the discrete-time domain, event-based consensus algorithms must be carefully designed especially in the resilient case. We can construct another resilient consensus algorithm motivated by the structures found in [75; 88], which deal with continuous-time multi-agent systems, as

$$x_i(k+1) = x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) \left(\hat{x}_j(k) - \hat{x}_i(k)\right). \tag{3.22}$$

The modification may be minor as the only difference is that $\hat{x}_i(k)$ is used instead of $x_i(k)$ in the second term of the right-hand side. Compared with Protocol 1, to guarantee the consensus error level of c, the choice of c_0 must be half as $c_0 \leq \gamma^{|\mathcal{R}|}/8|\mathcal{R}|$, which may increase the communication load. These results can be obtained by following a proof similar to that of Theorem 3.2.1.

In the next section, we present yet another protocol by further changing the terms in the update rule.

3.3 Protocol 2 for event-based consensus

In this section, we provide our second resilient consensus algorithm, referred to as Protocol 2.

To this end, we modify the update rule (3.4) in a way different from the protocol (3.22) discussed in Remark 3.2.1. It is pointed out that in Protocol 1, for obtaining the new state $x_i(k+1)$ of agent i, its own data appears only through the current state $x_i(k)$. On the one hand, this means that even when the new state is not communicated, it still needs to be stored at every time step. On the other, as the current state $x_i(k)$ is newer than $\hat{x}_i(k)$, it seems desirable for

speeding up the convergence. We will however show that it may be better to use only $\hat{x}_i(k)$ for both storage and convergence reasons. The protocol introduced below is motivated by those in [33; 88].

In the local update, for $k \in \mathbb{Z}_+$, every regular node $i \in \mathcal{R}$ updates its current state by

$$x_i(k+1) = \hat{x}_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) \left(\hat{x}_j(k) - \hat{x}_i(k)\right). \tag{3.23}$$

Note that the new state $x_i(k+1)$ need not be stored until the next time step, but is merely used for checking the condition of the triggering function $f_i(k)$ in (3.3). Accordingly, in the E-MSR, steps 1 and 2 should be adjusted so that agent i uses $\hat{x}_i(k)$ instead of $x_i(k)$ in determining the neighbor set $\mathcal{M}_i(k)$. The Comparison of storage period for current value $x_i(k)$ is shown in Fig. 3.1. It is Clear that protocol 2 requires less storage.

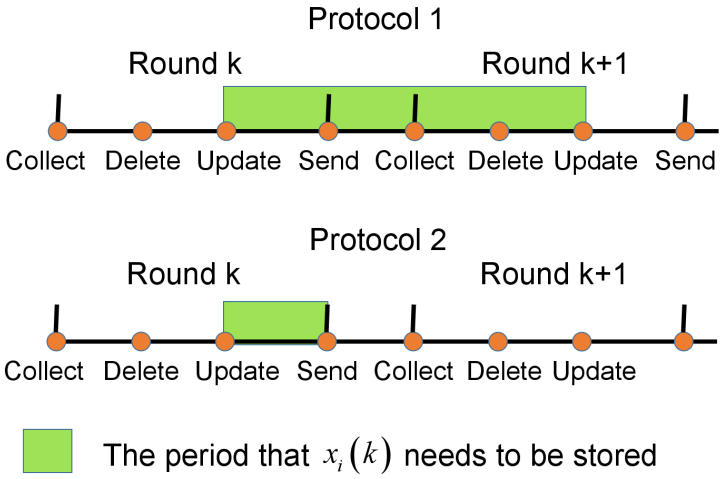


Figure 3.1: Storage period for $x_i(k)$

Then we are ready to present our second main result of this chapter, which is regarding Protocol 2.

Theorem 3.3.1. Under the F-total malicious model, the normal agents with E-MSR using (3.23) and (3.5) reach resilient consensus if and only if the underlying

graph is $(F+1, F+1)$-robust. The safety interval is given by $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$, and the consensus error level c is achieved if the parameter c_0 in the triggering function (3.3) satisfies

$$c_0 \leq \frac{\gamma^{|\mathcal{R}|-1}(1-\gamma)c}{1-\gamma^{|\mathcal{R}|-1}}. \tag{3.24}$$

Proof. The necessity part follows similar lines as those in the proof of Theorem 3.2.1. In the following, we thus give the sufficiency part.

First, we establish the safety condition in the sense of $x_i(k), \hat{x}_i(k) \in \mathcal{S}$ for regular nodes i. This is done by induction. At $k=0$, for each $i \in \mathcal{R}$, it holds $x_i(0), \hat{x}_i(0) \in \mathcal{S}$ by definition. Next, assume that at time k, we have $x_i(k), \hat{x}_i(k) \in \mathcal{S}$ for $i \in \mathcal{R}$. Then, for agent i, its neighbors $j \in \mathcal{M}_i(k)$ satisfy $\hat{x}_j(k) \in \mathcal{S}$ since there are at most F agents with values outside $\mathcal{S}$, and they are ignored in step 2 of the E-MSR. From the update rule (3.23), we have

$$\begin{aligned} x_i(k+1) &= a_{ii}(k)\hat{x}_i(k) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\hat{x}_j(k) \\ &\leq a_{ii}(k)\overline{\hat{x}}(k) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\overline{\hat{x}}(k) = \overline{\hat{x}}(k), \end{aligned} \tag{3.25}$$

where $a_{ii}(k) = 1 - \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)$. The inequality (3.25) means that the upper bound of every regular node is nonincreasing. Similarly, we have $x_i(k+1) \geq \underline{\hat{x}}(k)$, so we obtain $x_i(k) \in \mathcal{S}$ for $k \geq 0$. Furthermore, by (3.5), it holds that $\hat{x}_i(k+1) \in \mathcal{S}$. Hence, we have $\mathcal{S}$ as the safety interval.

For the consensus condition part, we first sort the regular communicated values $\hat{x}_i(k), i \in \mathcal{R}$, at time k in the entire graph from the smallest to the largest. Denote by $s_i(k)$ the index of the agent taking the ith value from the smallest. Hence, the values are sorted as $\hat{x}_{s_1}(k) \leq \hat{x}_{s_2}(k) \leq \cdots \leq \hat{x}_{s_{|\mathcal{R}|}}(k)$.

Introduce two sequences of conditions for the relation of each gap between two nodes. The first is given from below as

- A_1: $\hat{x}_{s_2}(k) - \hat{x}_{s_1}(k) \leq (c_0 + c_1 \mathrm{e}^{-\alpha k})/\gamma$,
- A_2: $\hat{x}_{s_3}(k) - \hat{x}_{s_2}(k) \leq (c_0 + c_1 \mathrm{e}^{-\alpha k})/\gamma^2$,
- $\cdots$
- A_{N-1}: $\hat{x}_{s_{|\mathcal{R}|}}(k) - \hat{x}_{s_{|\mathcal{R}|-1}}(k) \leq (c_0 + c_1 \mathrm{e}^{-\alpha k})/\gamma^{|\mathcal{R}|-1}$.

The other sequence is from above as

- B_N: $\hat{x}_{s_{|\mathcal{R}|}}(k) - \hat{x}_{s_{|\mathcal{R}|-1}}(k) \leq (c_0 + c_1 \mathrm{e}^{-\alpha k})/\gamma$,
- B_{N-1}: $\hat{x}_{s_{|\mathcal{R}|-1}}(k) - \hat{x}_{s_{|\mathcal{R}|-2}}(k) \leq (c_0 + c_1 \mathrm{e}^{-\alpha k})/\gamma^2$,
- $\cdots$
- B_2: $\hat{x}_{s_2}(k) - \hat{x}_{s_1}(k) \leq (c_0 + c_1 \mathrm{e}^{-\alpha k})/\gamma^{|\mathcal{R}|-1}$.

Denote by j_A the minimum j, $1 \leq j \leq |\mathcal{R}| - 1$, such that the condition A_j is not satisfied. Also, denote by j_B the maximum j, $2 \leq j \leq |\mathcal{R}|$, such that the condition B_j is not satisfied. Thus we have

$$\begin{aligned} \hat{x}_{s_{j_A+1}}(k) - \hat{x}_{s_{j_A}}(k) &> \frac{c_0 + c_1 \mathrm{e}^{-\alpha k}}{\gamma^{j_A}}, \\ \hat{x}_{s_{j_B}}(k) - \hat{x}_{s_{j_B-1}}(k) &> \frac{c_0 + c_1 \mathrm{e}^{-\alpha k}}{\gamma^{|\mathcal{R}|-j_B+1}}. \end{aligned} \tag{3.26}$$

Moreover, the conditions A_1 to A_{j_A-1} and B_{j_B+1} to B_N are satisfied. Then, for $0 \leq k \leq k'$, we introduce two sets

$$\begin{aligned} \mathcal{X}_1(k,k') &= \left\{ j \in \mathcal{V} : \ \hat{x}_j(k') < \hat{x}_{s_{j_A}}(k) + c_0 + c_1 \mathrm{e}^{-\alpha k} \right\}, \\ \mathcal{X}_2(k,k') &= \left\{ j \in \mathcal{V} : \ \hat{x}_j(k') > \hat{x}_{s_{j_B}}(k) - c_0 - c_1 \mathrm{e}^{-\alpha k} \right\}. \end{aligned}$$

There are two cases concerning the relationship between j_A and j_B. We study them separately below.

Case 1: $j_A < j_B$. Let the two subsets of the regular nodes be $\mathcal{V}_1 = \{s_1(k), s_2(k), \ldots, s_{j_A}(k)\}$ and $\mathcal{V}_2 = \{s_{j_B}(k), \ldots, s_{|\mathcal{R}|}(k)\}$. Note that all nodes in $\mathcal{V}_1$ are inside $\mathcal{X}_1(k,k)$, and those in $\mathcal{V}_2$ are inside $\mathcal{X}_2(k,k)$. Hence, $\mathcal{X}_1(k,k)$ and $\mathcal{X}_2(k,k)$ are nonempty. They are moreover disjoint. This is because by using the two inequalities in (3.26), from $1 \leq j_A < j_B \leq N$ and $0 < \gamma \leq 1/2$, it follows that

$$\begin{aligned}\hat{x}_{s_{j_B}}(k) - \hat{x}_{s_{j_A}}(k) &> \max\left\{\frac{1}{\gamma^{j_A}}, \frac{1}{\gamma^{|\mathcal{R}|-j_B+1}}\right\}\left(c_0 + c_1\mathrm{e}^{-\alpha k}\right) \\ &\geq 2\left(c_0 + c_1\mathrm{e}^{-\alpha k}\right).\end{aligned}$$

Thus, the $(F+1, F+1)$-robust graph guarantees that some regular node i in $\mathcal{X}_1(k,k)$ or $\mathcal{X}_2(k,k)$ has at least $F+1$ neighbors outside. We suppose that $i \in \mathcal{X}_1(k,k)$. By (3.23),

$$x_i(k+1) = a_{ii}(k)\hat{x}_i(k) + \sum_{j \in \mathcal{M}_i(k) \cap \mathcal{X}_1} a_{ij}(k)\hat{x}_j(k) + \sum_{j \in \mathcal{M}_i(k) \setminus \mathcal{X}_1} a_{ij}(k)\hat{x}_j(k),$$

where the simplified notation $\mathcal{X}_1$ is used for $\mathcal{X}_1(k,k)$. Since $\mathcal{M}_i(k) \setminus \mathcal{X}_1(k,k)$ is not empty, we have

$$x_i(k+1) \geq (1-\gamma)\hat{x}_{s_1}(k) + \gamma\hat{x}_{s_{j_A+1}}(k). \tag{3.27}$$

Using the conditions A_1 to A_{j_A-1}, we can bound $\hat{x}_{s_1}(k)$ from below as

$$\begin{aligned}\hat{x}_{s_1}(k) &\geq \hat{x}_{s_2}(k) - \frac{c_0 + c_1\mathrm{e}^{-\alpha k}}{\gamma} \\ &\geq \hat{x}_{s_3}(k) - \left(\frac{1}{\gamma} + \frac{1}{\gamma^2}\right)\left(c_0 + c_1\mathrm{e}^{-\alpha k}\right) \geq \cdots \\ &\geq \hat{x}_{s_{j_A}}(k) - \left(\frac{1}{\gamma} + \frac{1}{\gamma^2} + \cdots + \frac{1}{\gamma^{j_A-1}}\right)\left(c_0 + c_1\mathrm{e}^{-\alpha k}\right).\end{aligned}$$

Substitute this into (3.27) and obtain

$$\begin{aligned} x_i(k+1) &\geq \hat{x}_{s_{j_A}}(k) + \gamma\left(\hat{x}_{s_{j_A+1}}(k) - \hat{x}_{s_{j_A}}(k)\right) \\ &\quad - \frac{1}{\gamma^{j_A-1}}(c_0 + c_1\mathrm{e}^{-\alpha k}) + (c_0 + c_1\mathrm{e}^{-\alpha k}) \\ &> \hat{x}_{s_{j_A}}(k) + \gamma\frac{c_0 + c_1\mathrm{e}^{-\alpha k}}{\gamma^{j_A}} \\ &\quad - \frac{1}{\gamma^{j_A-1}}(c_0 + c_1\mathrm{e}^{-\alpha k}) + (c_0 + c_1\mathrm{e}^{-\alpha k}) \\ &= \hat{x}_{s_{j_A}}(k) + (c_0 + c_1\mathrm{e}^{-\alpha k}), \end{aligned} \tag{3.28}$$

where the second inequality follows by (3.26). Thus, this node i is moved out of set $\mathcal{X}_1(k, k+1)$ at time $k+1$.

We next show that the regular nodes not in $\mathcal{X}_1(k,k)$ at time k will not move in $\mathcal{X}_1(k, k+1)$ at time $k+1$. If node j has some neighbors inside $\mathcal{X}_1(k,k)$, then (3.27) and (3.28) hold and we know that the node does not move in $\mathcal{X}_1(k, k+1)$. If node j has neighbors only in $\mathcal{V} \setminus \mathcal{X}_1(k,k)$, then we have

$$x_j(k+1) \geq \hat{x}_{s_{j_A+1}}(k) > \hat{x}_{s_{j_A}}(k) + \frac{c_0 + c_1\mathrm{e}^{-\alpha k}}{\gamma^{j_A}}.$$

Clearly, node j does not move in $\mathcal{X}_1(k, k+1)$ in this case.

Therefore, the regular nodes in $\mathcal{X}_1(k, k+1)$ decrease in number as

$$\mathcal{X}_1(k, k+1) \cap \mathcal{R} \subsetneq \underline{\mathcal{X}}_1(k,k) \cap \mathcal{R}.$$

Similar results also hold if $i \in \mathcal{X}_2(k,k)$, and we have $\hat{x}_i(k+1)$ decreases more than $c_0 + c_1\mathrm{e}^{-\alpha k}$ compared with $\hat{x}_{s_{j_B}}(k)$.

As a result, if the conditions A_{j_A} and B_{j_B} with $j_A < j_B$ are not satisfied, after $|\mathcal{R}|$ steps, the set $\mathcal{X}_1(k, k+|\mathcal{R}|)$ or $\mathcal{X}_2(k, k+|\mathcal{R}|)$ becomes empty in regular nodes. It then follows that $\underline{\hat{x}}(k+|\mathcal{R}|)$ increases more than $c_0 + c_1\mathrm{e}^{-\alpha k}$ or $\overline{\hat{x}}(k+|\mathcal{R}|)$

decreases more than $c_0 + c_1 e^{-\alpha k}$.

A special case in Case 1 is when $j_A = j_B - 1$. It corresponds to having only one pair of nodes whose difference in values does not satisfy the condition. By applying a similar analysis, we have that $\underline{\hat{x}}(k+|\mathcal{R}|)$ increases more than $c_0+c_1 e^{-\alpha k}$ or $\overline{\hat{x}}(k+|\mathcal{R}|)$ decreases more than $c_0 + c_1 e^{-\alpha|\mathcal{R}|}$.

Case 2: $j_A \geq j_B$. This case is impossible. We can show this by contradiction as follows. Since $j_A \geq j_B$, we know that A_{j_B-1} and B_{j_A+1} are both satisfied. Combined with A_{j_A} and B_{j_B} not being satisfied, we have

$$\frac{c_0 + c_1 e^{-\alpha k}}{\gamma^{|\mathcal{R}|-j_B+1}} < \hat{x}_{s_{j_B}}(k) - \hat{x}_{s_{j_B-1}}(k) \leq \frac{c_0 + c_1 e^{-\alpha k}}{\gamma^{j_B-1}}, \tag{3.29}$$

$$\frac{c_0 + c_1 e^{-\alpha k}}{\gamma^{j_A}} < \hat{x}_{s_{j_A+1}}(k) - \hat{x}_{s_{j_A}}(k) \leq \frac{c_0 + c_1 e^{-\alpha k}}{\gamma^{|\mathcal{R}|-j_A}}. \tag{3.30}$$

The inequalities in the first relations in (3.29) indicate that it must hold $j_B > (|\mathcal{R}| + 1)/2$. The second set of inequalities in (3.30) also implies $j_A < |\mathcal{R}|/2$. Consequently, we have $j_A < j_B$, which is in contradiction with $j_A \geq j_B$.

We can now conclude that after a finite number of time steps, all conditions from A_1 to A_m and B_{m+2} to $B_{|\mathcal{R}|}$, where $0 \leq m \leq |\mathcal{R}| - 1$, must be satisfied. Otherwise the difference between $\overline{\hat{x}}(k)$ and $\underline{\hat{x}}(k)$ will decrease more than c_0 by an update induced by an event. From the analysis for the safety condition, we know that $\overline{\hat{x}}(k)$ is nonincreasing and $\underline{\hat{x}}(k)$ is nondecreasing. Hence, if the events continuously occur, $\overline{\hat{x}}(k) - \underline{\hat{x}}(k)$ will become smaller and eventually negative, which cannot happen. This completes the proof. □

Protocol 2 enables us to achieve resilient consensus with data communicated via event-based protocols. We can see by directly comparing with the result for Protocol 1 that the bound obtained here for the parameter c_0 is larger, indicating that it is less conservative. Hence, to achieve the same level c of consensus error, we may use a larger c_0 in Protocol 2, which will result in less frequent

transmissions. We confirm this property later in Section 3.4 through numerical simulations.

A unique aspect of Protocol 2 is that the proof technique used in Theorem 3.3.1 is different from those used in the recent works such as [16; 17; 18; 44; 45] and also in the proof of Theorem 3.2.1. The conventional technique could be employed here, but this will result in the same bound on c_0 as in Theorem 3.2.1. In fact, as we see below, the bound obtained in Theorem 3.3.1 is tight for some graphs.

Remark 3.3.1. We present an example of a multi-agent system whose error in consensus among the agents is equal to the bound obtained in Theorem 3.3.1. Such a graph may be called a worst-case graph. Consider the network in Fig. 3.2 with four nodes which are all regular and thus $F = 0$. Note that the graph contains a directed spanning tree. The initial values $x_i(0)$ of the nodes and the (constant) weights $a_{ij}(k)$ on the edges are indicated in the figure. Since the weights are all $1/2$ (and thus $\gamma = 1/2$), for nodes having two neighbors, their own values are not used in the update rule (3.23). Moreover, for the node in the far left, a self-loop is shown to indicate that this node uses its own value. The node in the far right has no incoming edge, and thus its value will not change over time.

By setting the parameters for the triggering function as $c_0 = 1$ and $c_1 = 0$, it follows that there will be no event at any time. The difference in their values is 14, which can be obtained as c by equating the inequality (3.24) in Theorem 3.3.1. In comparison, for Protocol 1, the bound c on the difference will be 256 by Theorem 3.2.1; this is much larger, indicating the conservatism of the approach.

Note that the graph structure in Fig. 3.2 is obtained based on the proof of Theorem 3.3.1. It is a bit special in the sense that not all agents have self-loops. To comply with the theory, we can extend this example by adding self-loops; it

will not be a worst-case graph any longer, but the difference in the values will be larger than other graphs.

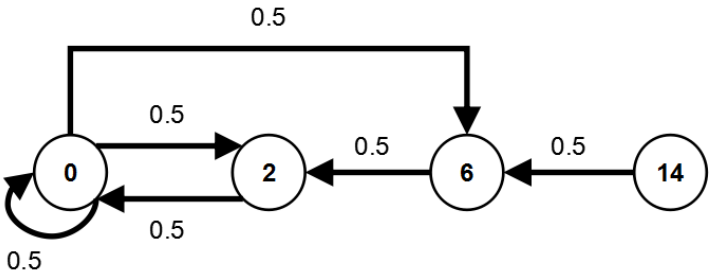


Figure 3.2: Worst-case graph with $|\mathcal{R}|$=4

3.4 Numerical example

In this section, we illustrate the proposed resilient protocols via numerical simulations. We first examine a small-scale network and then focus on the scalability for larger systems.

3.4.1 Small network

We consider the multi-agent system with seven nodes whose connectivity graph is shown in Fig. 2.1; as already mentioned, this graph is $(3, 3)$-robust. We compare the performance of Protocols 1 and 2 using different parameters in event-triggering. In particular, we test the two cases of $c_0 > 0$ and $c_0 = 0$. Here, nodes 5 and 7 are set to behave maliciously by continuously oscillating their values; in all simulations, we used the same state values for them. The initial state was chosen the same for each run as well at $x(0) = [1\ 2\ 3\ 5\ 4\ 6\ 4]^T$. We also took $\gamma = 0.3$.

First, we examine the case of $c_0 > 0$. We fixed the consensus error bound as $c = 1$. For Protocol 1, based on Theorem 3.2.1, we chose $c_0 = 1.22 \times 10^{-4}$. The remaining parameters were selected as $c_1 = 0.5$ and $\alpha = 0.03$. The time

responses are shown in Fig. 3.3, where the x-axis represents the sampling time k, and the y-axis the values of the agents. Moreover, the time instants when each node makes a broadcast are shown by the markers $\bullet$ in the color corresponding to that of its time response curve. On the other hand, for Protocol 2, we chose $c_0 = 5.72 \times 10^{-3}$ according to Theorem 3.3.1, and other parameters were taken as above with $c_1 = 0.5$ and $\alpha = 0.03$. The time responses of Protocol 2 are plotted in Fig. 3.4.

We observe that both protocols managed to achieve the desired level of consensus specified by $c = 1$ based on event-triggered communication. Moreover, there is very little sign of being influenced by the behavior of the malicious nodes. In fact, for Protocol 1, after 600 steps, the consensus error among the regular nodes became 5.24×10^{-5}, with 5.4 times of transmissions on average for the regular nodes. On the other hands, for Protocol 2, the consensus error was 8.63×10^{-3}, with 4.6 times of transmissions on average. Thus, we confirm that Protocol 2 is less conservative for the given $c = 1$.

Next, by setting $c_0 = 0$, we demonstrate that exact resilient consensus can be attained while reducing the number of transmissions. To this end, for both protocols, we set $c_1 = 0.5$ and $\alpha = 0.03$ as in the previous simulations. In this case, the threshold that determines the timings of events eventually goes to zero (due to $c_0 = 0$).

The time responses of the two protocols are shown in Figs. 3.5 and 3.6. For Protocol 1, after 600 steps, the consensus errors among the regular nodes became essentially zero at 5.71×10^{-9}, where the average number of triggering times for the regular nodes is 10. Similarly, for Protocol 2, the consensus error at time $k = 600$ was 1.73×10^{-8} with 12.4 triggering times on average per regular node. Protocol 1 is particularly impressive in terms of the limited amount of communication. In contrast, for Protocol 2, information exchange among the

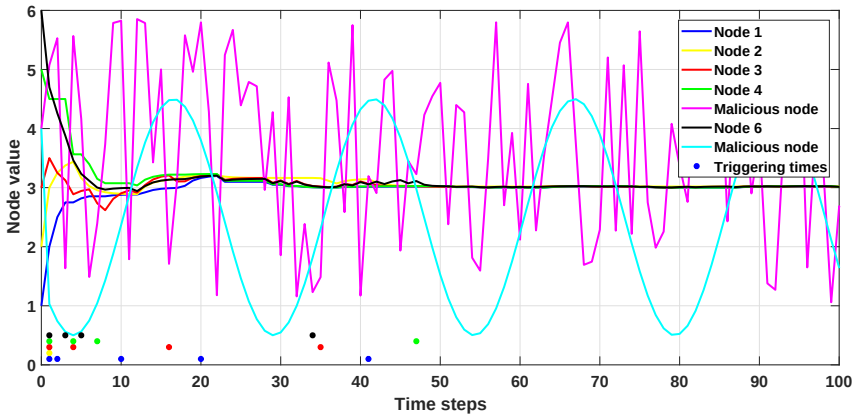


Figure 3.3: Protocol 1 with $c_0 = 1.215 \times 10^{-4}$, $c_1 = 0.5$, and $\alpha = 0.03$

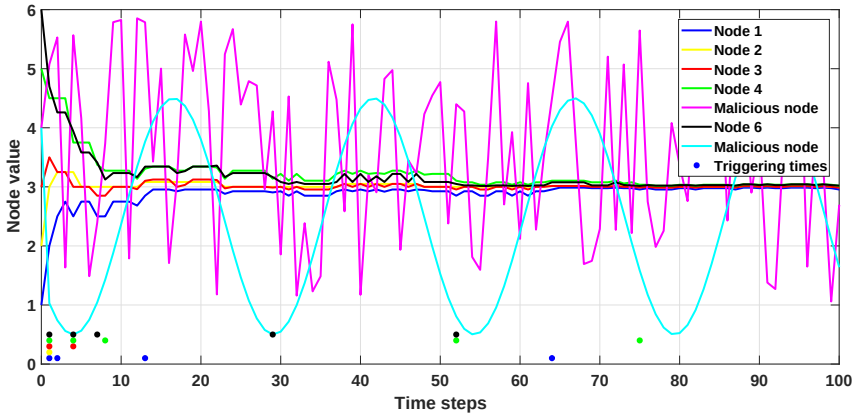


Figure 3.4: Protocol 2 with $c_0 = 5.72 \times 10^{-3}$, $c_1 = 0.5$, and $\alpha = 0.03$

nodes takes place for a longer time.

Further comparisons were made by implementing time-triggering communication in both protocols. Periodic transmissions are made so that after 600 time steps, the regular nodes make the same number of triggering times as those in the event-triggered case with $c_0 = 0$ above. This means that for Protocol 1, each node transmits every 60 steps and for Protocol 2 every 50 steps. At time $k = 600$, the consensus error was 5.04×10^{-8} for Protocol 1 and 5.80×10^{-3} for Protocol 2. It is clear that under both protocols, the event-triggered schemes perform better. Their time responses are shown in Figs. 3.7 and 3.8. Due to the periodic transmission, the convergence is slow and the responses between the transmission

times are oscillatory.

3.4.2 Scalability of the proposed approach

In this part, we carry out a number of simulations to check the scalability of the proposed protocols using large-scale networks. In particular, we focus on how the number of transmissions can be kept low even if the numbers of neighbors and even the malicious ones are large. As in the previous simulations, the two cases of $c_0 > 0$ and $c_0 = 0$ are examined and compare with the time-triggered case.

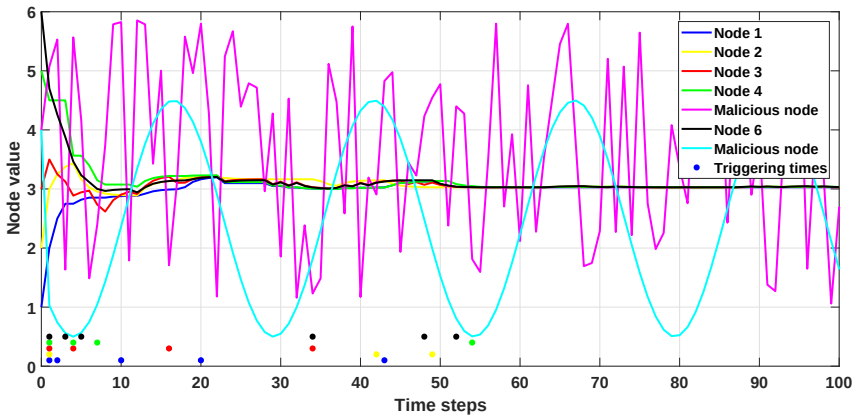


Figure 3.5: Protocol 1 with $c_0 = 0$, $c_1 = 0.5$, and $\alpha = 0.03$

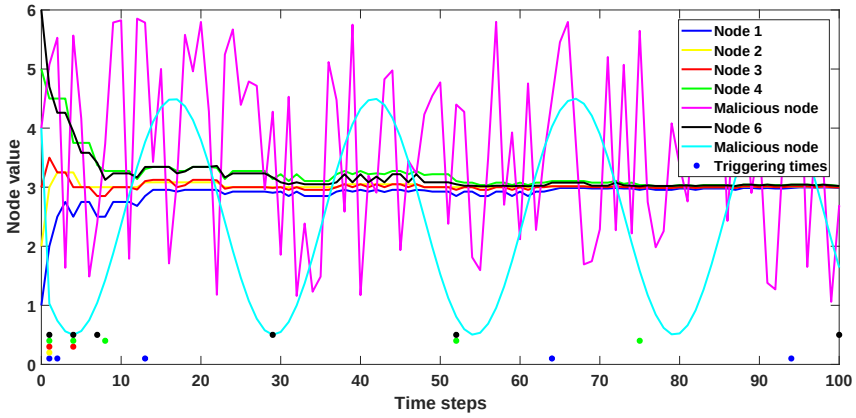


Figure 3.6: Protocol 2 with $c_0 = 0$, $c_1 = 0.5$, and $\alpha = 0.03$

We employ three complete graphs with 10 nodes, 50 nodes, and 100 nodes.

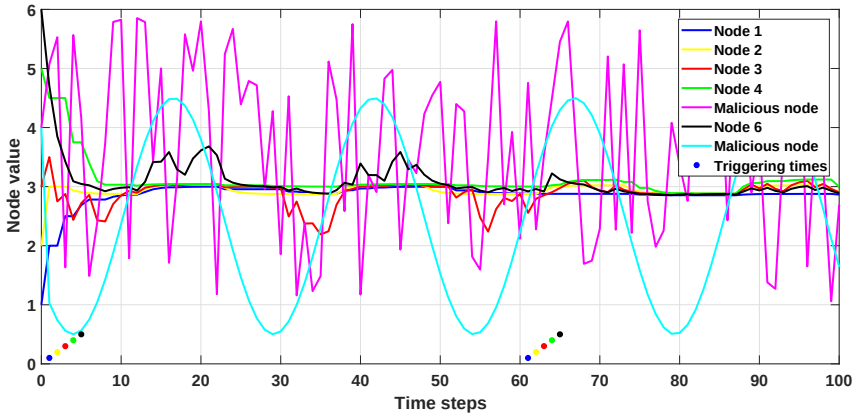


Figure 3.7: Protocol 1 under periodic communication with period 60

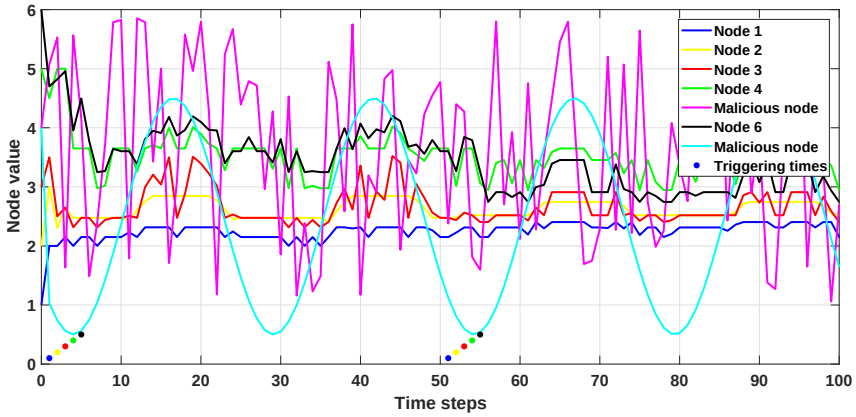


Figure 3.8: Protocol 2 under periodic communication with period 50

By Lemma 2.2.1, we know that a 10-node complete graph is $(5, 5)$-robust. Thus, we introduce four malicious nodes. Similarly, in the 50- and 100-node cases, we set 24 and 49 nodes to be malicious, respectively.

The first case is with $c_0 > 0$. In particular, for both Protocols 1 and 2, we chose $c_0 = 0.1$, $c_1 = 1$, and $\alpha = 2$. For each graph, we performed Monte Carlo simulations for 100 runs by randomly taking initial states under uniform distribution between 0 and 100. Each agent made updates until the consensus error becomes 0.01 for Protocol 1 and 0.3 for Protocol 2. The performance of Protocols 1 and 2 is displayed in Tables 3.1 and 3.2 in terms of the average number of triggering times per regular node.

Table 3.1: Protocol 1 with consensus error 0.01

Graphs	Event-Triggered		Time-Triggered
	$c_0 = 0.1$	$c_0 = 0$	
10 nodes	4.9	4.4	9.8
50 nodes	6.5	5.4	11.4
100 nodes	7.1	5.7	11.9

Table 3.2: Protocol 2 with consensus error 0.3

Graphs	Event-Triggered		Time-Triggered
	$c_0 = 0.1$	$c_0 = 0$	
10 nodes	4.7	3.8	6.9
50 nodes	5.9	5.6	8.1
100 nodes	6.2	6.5	8.4

It is noticed that in general, as the number of agents increases, triggering times increase only mildly to reach the same consensus error for both protocols. There is a slight difference in the performance between the protocols as discussed after Theorem 3.3.1. In particular, for the same size of c_0, Protocol 1 achieves smaller error than Protocol 2.

We proceed to the second case with $c_0 = 0$. Specifically, for Protocol 1, we used $c_0 = 0$, $c_1 = 0.5$, and $\alpha = 0.05$. For Protocol 2, we used the same c_0 and c_1, but a smaller $\alpha = 0.01$. The results are summarized in the same tables. Compared to the case with $c_0 > 0$, we observe that the triggering times are similar, though the scalability may be less in that the triggering times increase as the graph sizes increase.

Finally, in the two tables, we display the number of triggering times for the time-triggered case, where every node transmits its value at every time step. It is evident that the event-triggered case performs much better for both protocols. From these simulations, we can conclude that the event-based protocols can efficiently eliminate the amount of communications.

3.4.3 The effects of triggering parameters

In this part, we observe the effects of triggering parameters c_0, c_1 and α through a comparison simulation in Protocol 1. The first set of parameters are chosen as $c_0 = 0.1, c_1 = 0.5, \alpha = 0.03$. The second set of parameters are chosen as $c_0 = 0.1, c_1 = 0.5, \alpha = 0.03$. The time responses are plotted in Fig 3.9 and Fig 3.10. We can see that with positive c_0, the triggering points are effectively reduced, and it stops to communicate with each other when the regular agents are close. If $c_0 = 0$, then the triggering points keep happen in a long time period. From another viewpoint, the error level with positive c_0 cannot decrease to 0. However, the error level with $c_0 = 0$ can zero a zero error level when time goes to infinity. The role of c_0 can be seen as a asymptotic tolerance level, and the role of c_1 and α can be seen as tolerance reduction factor.

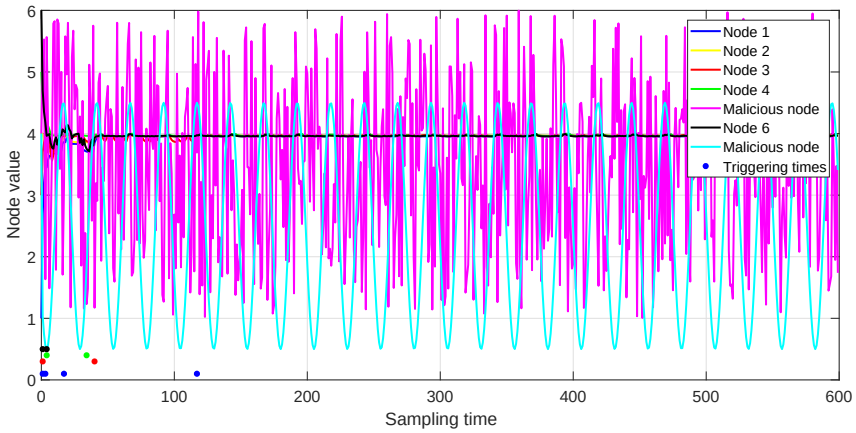


Figure 3.9: Time responses with $c_0 = 0.1, c_1 = 0.5, \alpha = 0.03$ in Protocol 1

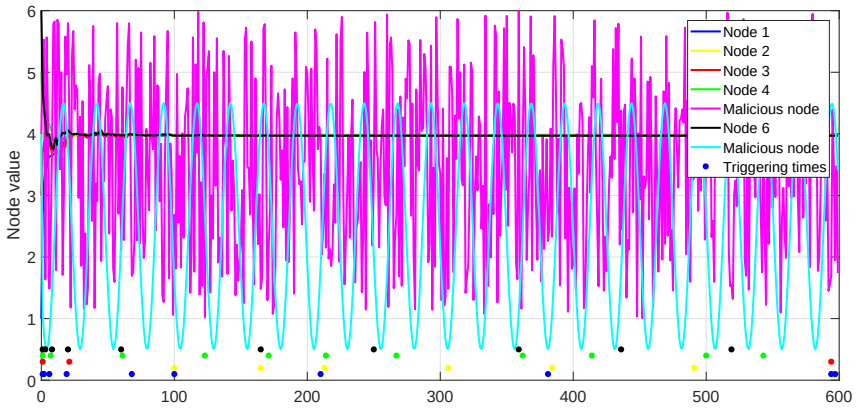


Figure 3.10: Time responses with $c_0 = 0, c_1 = 0.5, \alpha = 0.03$ in Protocol 1

Chapter 4

An Event-Triggered Approach to Quantized Resilient Consensus

In this chapter, we consider an event-triggered update scheme for the problem of multi-agent consensus in the presence of faulty and malicious agents within the network. In particular, we focus on the case where the agents take integer (or quantized) values. To keep the regular agents from being affected by the behavior of faulty agents, algorithms of the mean subsequence reduced (MSR) type are employed, where neighbors taking extreme values are ignored in the updates. Different from the real-valued case, the quantized version requires the update rule to be randomized. We characterize the error bound on the achievable level of consensus among the agents as well as the necessary structure for the network in terms of the notion of robust graphs. We verify via a numerical example the effectiveness of the proposed algorithms. This part is published in [84].

4.1 Problem formulation

The quantized event-based consensus protocol can be outlined as follows: At each discrete-time instant $k \in \mathbb{Z}_+$, each node makes an update in its state value $x_i(k)$. It decides to broadcast its current value to its neighbors depends on the triggering function. More concretely, each node i is equipped with an auxiliary variable $\hat{x}_i(k)$ which stores the last communicated value before time k. This is compared with the updated state $x_i(k+1)$. If the state has changed sufficiently, then it will be sent to its neighbors and the auxiliary variable will be replaced.

We employ the following quantized event-based update rule for the multi-agent system:

$$x_i(k+1) = x_i(k) + Q\left(\sum_{j\in\mathcal{N}_i} a_{ij}(k)\left(\hat{x}_j(k) - x_i(k)\right)\right),$$

where $x_i(k) \in \mathbb{Z}$ is the state of node i, and $\hat{x}_j(k) \in \mathbb{Z}$ is the last communicated state of node j at time k. The latter is defined as $\hat{x}_j(k) = x_j(t_l^j)$, $k \in [t_l^j, t_{l+1}^j)$, where $t_0^j, t_1^j, \ldots$ denote the transmission times determined by the triggering function to be given below. The initial values $x_i(0)$, $\hat{x}_j(0)$ are given. The weight $a_{ij}(k)$ for the edge $(j,i) \in \mathcal{E}$ satisfies $\gamma \leq a_{ij}(k) < 1$ or $a_{ij}(k) = 0$, and $\sum_{j\in\mathcal{N}_i(k)} a_{ij}(k) = 1$, where $\gamma > 0$ is the lower bound of the weights.

Here, $Q(y)$ denotes the probabilistic quantizer function. It is given by

$$Q(y) = \begin{cases} \lfloor y \rfloor & \text{with probability } p(y), \\ \lceil y \rceil & \text{with probability } 1 - p(y), \end{cases}$$

where $p(y) = \lceil y \rceil - y$.

We note that the the quantizer has to be chosen carefully and neither floor nor ceil quantizer can guarantee resilient consensus. In Fig. 4.1, we provide an

example to explain that in some cases, floor or ceil quantizer cannot reach resilient consensus.

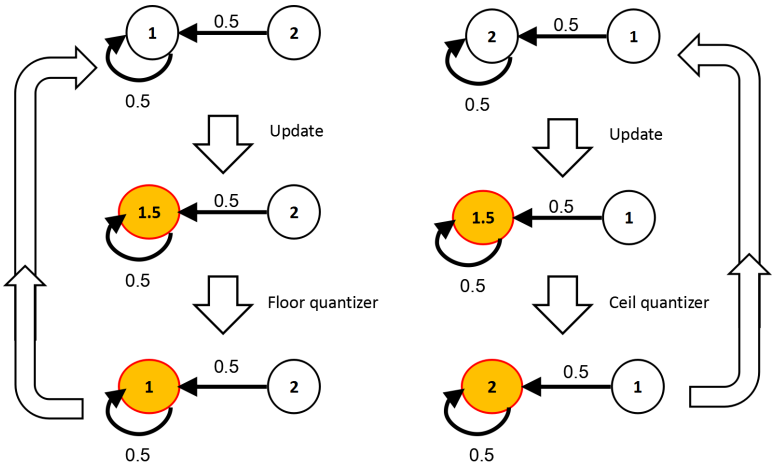


Figure 4.1: Example of floor and ceil quantizer

Denote the error between the updated state $x_i(k+1)$ and the last communicated state $\hat{x}_i(k)$ by $e_i(k+1) = \hat{x}_i(k) - x_i(k+1)$ with $e_i(0) = 0$. The triggering function is given as

$$f_i(k+1) = |e_i(k+1)| - \left\lfloor c_0 + c_1 \mathrm{e}^{-\alpha(k+1)} \right\rfloor, \tag{4.1}$$

where $c_0, c_1, \alpha > 0$ are positive constants.

Now, we introduce the notion of quantized resilient consensus for multi-agent systems.

Definition 4.1.1. *(*Quantized resilient consensus*) If for any possible sets and behaviors of the malicious agents and any state values of the regular nodes, the following two conditions are satisfied, then we say that the regular agents reach quantized resilient consensus almost surely:*

1. *Safety condition: There exists a bounded interval $\mathcal{S} \subset \mathbb{Z}$ determined by the initial states of regular agents such that $x(k) \in \mathcal{S}$ for all $i \in \mathcal{R}$, $k \in \mathbb{Z}_+$.*

2. *Consensus condition: There exists a finite time* $k_a > 0$ *with probability one such that* $x_i(k) \in \mathcal{C}_\beta$ *for all* $k > k_a$ *and* $i \in \mathcal{R}$*, where the approximate consensus set* $\mathcal{C}_\beta$ *for* $\beta \geq 0$ *is given by* $\mathcal{C}_\beta = \{x \in \mathbb{Z}^n : |x_j - x_i| \leq \beta\}$.

In this section, we would like to design event-based update rules for the regular agents to reach quantized resilient consensus under the F-total model by using only local information obtained from their neighbors.

The resilient consensus algorithms developed in this paper follows the basic approach from [16; 17; 45]. We present two resilient algorithms that can be seen as extensions of our recent work [85], which dealt with the real-valued states case. The main difference in the algorithms is that they are randomized due to the use of the probabilistic quantizer.

On the other hand, for the quantization in resilient consensus problem, our approach is based on [18], where the advantages and the necessity of such quantizers are discussed in detail. In fact, our study can be viewed as a generalization since in the case of $c_0 = c_1 = 0$, the proposed update rules coincide with the one without event-triggered protocols studied there.

4.2 Quantized resilient consensus protocol

In this section, we outline our first protocol to solve the quantized resilient consensus problem, which will be referred to as Protocol 1.

In this protocol, the regular nodes make updates at every time step in a synchronous manner, but only when an event happens, they update the auxiliary values and then send them to neighbors. This algorithm is called the Quantized Event-Based Mean Subsequence Reduced (QE-MSR) algorithm.

The QE-MSR algorithm can be described by the four steps as follows:

1. *(Collecting neighbors' information)* At each time step k, every regular node

$i \in \mathcal{R}$ uses the values $\hat{x}_j(k), j \in \mathcal{N}_i$ most recently communicated from the neighbors as well as its own value $x_i(k)$ and sorts them from the largest to the smallest.

2. *(Deleting suspicious values)* Comparing with $x_i(k)$, node i removes the F largest and F smallest values from its neighbors. If the number of values larger or smaller than $x_i(k)$ is less than F, then all of them are removed. The removed data is considered as suspicious and will not be used in the local update at this time step. The set of agents whose values remain is written by $\mathcal{M}_i(k) \subset \mathcal{N}_i$.

3. *(Local update)* Node i updates its state by

$$x_i(k+1) = x_i(k) + Q\left(\sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(\hat{x}_j(k) - x_i(k)\right)\right). \tag{4.2}$$

4. *(Communication update)* Node i checks its own triggering function if its own triggering function $f_i(k+1)$ in (4.1) is positive or not. Then, it sets $\hat{x}_i(k+1)$ as

$$\hat{x}_i\left(k+1\right) = \begin{cases} x_i(k+1) & \text{if } f_i(k+1) > 0, \\ \hat{x}_i(k) & \text{otherwise.} \end{cases} \tag{4.3}$$

The communication rule in this algorithm shows that only when the current value has sufficiently changed to exceed a threshold for the triggering function, then the auxiliary variable will be updated, and only at this time, the node sends its value to its neighbors. This event triggering scheme can significantly reduce the communication burden; we will see this in the numerical example in Section 4.4.

We are now ready to present our main result for Protocol 1 of this section. Let

the interval $\mathcal{S}$ be given as $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$, where $\underline{\hat{x}}\,(k) = \min_{i\in\mathcal{R}}\{x_i(k), \hat{x}_i(k)\}$ and $\overline{\hat{x}}\,(k) = \max_{i\in\mathcal{R}}\{x_i(k), \hat{x}_i(k)\}$.

Theorem 4.2.1. *Under the F-total malicious model, the regular agents with the QE-MSR using* (4.2) *and* (4.3) *reach quantized resilient consensus in finite time almost surely if and only if the underlying graph is $(F+1, F+1)$-robust. The safety interval is given by $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$, and the approximate consensus set $\mathcal{C}_\beta$ is given with*

$$\beta = \min\left\{ |\mathcal{S}|\,, \left\lfloor 2c_0\left(\frac{2|\mathcal{R}|}{\gamma^{|\mathcal{R}|}} + 1\right)\right\rfloor\right\}, \tag{4.4}$$

where $|\mathcal{S}| = \overline{\hat{x}}(0) - \underline{\hat{x}}(0)$.

The proof of this theorem relies on the technical result from [36] (Theorem 2). We present it here as a lemma with minor modifications adapted for our problem setup.

Lemma 4.2.1. *Consider the network of agents interacting over the graph $\mathcal{G}$ through the QE-MSR algorithm. Suppose that the following three conditions are satisfied for the regular agents:*

(C1) $x_i(k), \hat{x}_i(k) \in \mathcal{S}$ *for all* $i \in \mathcal{R}$ *and* $k \in \mathbb{Z}_+$.

(C2) For each $x(k) = x_0$, $\hat{x}(k) = \hat{x}_0$, *there exists a finite time* k_x *such that* $Prob\{x(k+k_x), \hat{x}(k+k_x) \in C_\beta \mid x(k) = x_0, \hat{x}(k) = \hat{x}_0\} > 0$.

(C3) If $x(k), \hat{x}(k) \in C_\beta$, *then* $x(k'), \hat{x}(k') \in C_\beta$, $k' \geq k$.

Then, the regular agents reach quantized resilient consensus in finite time almost surely.

Proof of Theorem 4.2.1: (Necessity) This essentially follows from [18], which considers the special case without the triggering function, that is, $c_0 = c_1 = 0$.

(Sufficiency) We must show that the three conditions in Lemma 4.2.1 are met.

We first show (C1) by induction. Note that the update rule (4.2) can be expressed as follows:

$$x_i(k+1) = Q\left(\sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\hat{x}_j(k) + a_{ii}(k)x_i(k)\right), \tag{4.5}$$

where

$$a_{ii}(k) = 1 - \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k). \tag{4.6}$$

At time $k = 0$, it is clear that $\hat{x}_i(0), x_i(0) \in \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right], i \in \mathcal{R}$. The nodes in $\mathcal{M}_i(k)$ take values in $\mathcal{S}$, since the nodes outside $\mathcal{S}$ are at most F and will be removed by the QE-MSR. Then from (4.5), we have $x_i(k+1) \in \mathcal{S}$. Moreover, by (4.3), it follows $\hat{x}_i(k+1) \in \mathcal{S}$ and hence (C1) is satisfied with this interval $\mathcal{S}$. This implies that $\mathcal{C}_\beta \subset \mathcal{S}$ and thus $\beta \leq |\mathcal{S}|$.

We next check (C2). Note that for time k between two triggering instants, we have $f_i(k) \leq 0$. Moreover, for the neighbor $j \in \mathcal{N}_i$, if $f_j(k) > 0$, then $\hat{x}_j(k) = x_j(k)$ and otherwise $\hat{x}_j(k) = \hat{x}_j(k-1) = x_j(k) + e_j(k)$. Thus, by letting

$$\hat{e}_j(k) = \begin{cases} e_j(k) & \text{if } f_j(k) \leq 0, \\ 0 & \text{otherwise,} \end{cases}$$

it always holds from (4.1) that $\hat{x}_j(k) = x_j(k) + \hat{e}_j(k)$ and

$$|\hat{e}_j(k)| \leq c_0 + c_1 \mathrm{e}^{-\alpha k}, \quad \forall k \geq 0. \tag{4.7}$$

Then, from (4.5), due to the probabilistic quantization in (4.2), it holds with

positive probability,

$$x_i(k+1) = \left\lfloor a_{ii}(k)x_i(k) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)x_j(k) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k) \right\rfloor. \tag{4.8}$$

Let $\overline{x}(k) = \max_{i\in\mathcal{R}} x_i(k)$ and $\underline{x}(k) = \min_{i\in\mathcal{R}} x_i(k)$. Then, with positive probability, we have

$$\begin{aligned} x_i(k+1) &\leq \left\lfloor a_{ii}(k)\overline{x}(k) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\overline{x}(k) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k) \right\rfloor \\ &= \left\lfloor \overline{x}(k) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k) \right\rfloor \leq \left\lfloor \overline{x}(k) + \max_{j\in\mathcal{M}_i(k)} |\hat{e}_j(k)| \right\rfloor. \end{aligned}$$

Thus, by (4.7) it follows

$$x_i(k+1) \leq \left\lfloor \overline{x}(k) + c_0 + c_1 \mathrm{e}^{-\alpha k} \right\rfloor. \tag{4.9}$$

Let $V(k) = \overline{x}(k) - \underline{x}(k)$ and $\hat{V}(k) = \overline{\hat{x}}(k) - \underline{\hat{x}}(k)$. Then, we introduce two sequences given by

$$\overline{x}_0(k+1) = \overline{x}_0(k) + c_0 + c_1 \mathrm{e}^{-\alpha k}, \tag{4.10}$$

$$\underline{x}_0(k+1) = \underline{x}_0(k) - c_0 - c_1 \mathrm{e}^{-\alpha k}, \tag{4.11}$$

where $\overline{x}_0(0) = \overline{x}(0) - \sigma_0$, and $\underline{x}_0(0) = \underline{x}(0) + \sigma_0$ with $\sigma_0 = \sigma V(0)$. We next introduce another sequence $\varepsilon_0(k)$ defined by

$$\varepsilon_0(k+1) = \gamma\varepsilon_0(k) - (1-\gamma)\sigma_0, \tag{4.12}$$

where $\varepsilon_0(0) = \varepsilon V(0)$. Take the parameters $\varepsilon, \sigma > 0$ so that

$$\varepsilon + \sigma = \frac{1}{2}, \ \sigma < \frac{\gamma^N}{1-\gamma^N}\varepsilon. \tag{4.13}$$

For the sequence $\varepsilon_0(k)$, let

$$\begin{aligned}\overline{\mathcal{X}}_0(k, \varepsilon_0(k)) &= \{j \in \mathcal{V} : \ x_j(k) > \overline{x}_0(k) - \varepsilon_0(k)\}, \\ \underline{\mathcal{X}}_0(k, \varepsilon_0(k)) &= \{j \in \mathcal{V} : \ x_j(k) < \underline{x}_0(k) + \varepsilon_0(k)\}.\end{aligned}$$

In the following, we show that $\overline{\mathcal{X}_0}(k, \varepsilon_0(k))$ and $\underline{\mathcal{X}_0}(k, \varepsilon_0(k))$ are disjoint sets. To this end, we show

$$\overline{x}_0(k) - \varepsilon_0(k) > \underline{x}_0(k) + \varepsilon_0(k), \tag{4.14}$$

By the update rules of $\overline{x}_0(k)$ and $\underline{x}_0(k)$ in (4.10) and (4.11),

$$\begin{aligned}(\overline{x}_0(k) - \varepsilon_0(k)) - (\underline{x}_0(k) + \varepsilon_0(k)) = &\left(\overline{x}_0(0) + c_0 k + c_1 \frac{1 - \mathrm{e}^{-\alpha k}}{1 - \mathrm{e}^{-\alpha}}\right) \\ &- \left(\underline{x}_0(0) - c_0 k - c_1 \frac{1 - \mathrm{e}^{-\alpha k}}{1 - \mathrm{e}^{-\alpha}}\right) - 2\varepsilon_0(k).\end{aligned} \tag{4.15}$$

Then by substituting $\overline{x}_0(0) = \overline{x}(0) - \sigma_0$ and $\underline{x}_0(0) = \underline{x}(0) + \sigma_0$ into the right-hand side of (4.15), we obtain

$$\begin{aligned}&(\overline{x}_0(k) - \varepsilon_0(k)) - (\underline{x}_0(k) + \varepsilon_0(k)) \\ &\quad = (\overline{x}(0) - \underline{x}(0)) - 2\sigma_0 + 2c_0 k + 2c_1 \frac{1 - \mathrm{e}^{-\alpha k}}{1 - \mathrm{e}^{-\alpha}} - 2\varepsilon_0(k) \\ &\quad = V(0) - 2\sigma V(0) + 2c_0 k + 2c_1 \frac{1 - \mathrm{e}^{-\alpha k}}{1 - \mathrm{e}^{-\alpha}} - 2\varepsilon_0(k).\end{aligned} \tag{4.16}$$

Because of (4.12) and $0 < \gamma < 1$, we easily have that $\varepsilon_0(k+1) < \varepsilon_0(k)$, and hence $\varepsilon_0(k) < \varepsilon_0(0) = \varepsilon V(0)$. We thus obtain

$$(\overline{x}_0(k) - \varepsilon_0(k)) - (\underline{x}_0(k) + \varepsilon_0(k)) > (1 - 2\sigma - 2\varepsilon)V(0) + 2c_0 k + 2c_1 \frac{1 - \mathrm{e}^{-\alpha k}}{1 - \mathrm{e}^{-\alpha}} > 0,$$

where the last inequality holds since $\sigma + \varepsilon = 1/2$ from (4.13). Consequently, we have (4.14).

From the above, we have that the two sets $\overline{\mathcal{X}_0}(0, \varepsilon_0(0))$ and $\underline{\mathcal{X}_0}(0, \varepsilon_0(0))$ are nonempty with at least one regular node in each and moreover disjoint. Therefore, by the assumption of $(F+1,F+1)$-robustness, there are three cases:

1. All nodes in $\overline{\mathcal{X}_0}(0, \varepsilon_0(0))$ have $F+1$ neighbors or more from outside.
2. All nodes in $\underline{\mathcal{X}_0}(0, \varepsilon_0(0))$ have $F+1$ neighbors or more from outside.
3. The total number of nodes in $\overline{\mathcal{X}_0}(0, \varepsilon_0(0))$ and $\underline{\mathcal{X}_0}(0, \varepsilon_0(0))$ having $F+1$ neighbors or more from outside of its own set is no smaller than $F+1$.

Notice that in any of the three cases, there exists at least one regular agent $i \in \mathcal{R}$ in either $\overline{\mathcal{X}_0}(0, \varepsilon_0(0))$ or $\underline{\mathcal{X}_0}(0, \varepsilon_0(0))$ that has $F+1$ neighbors or more from outside of its own set. In the following, we suppose that this node i belongs to $\overline{\mathcal{X}_0}(0, \varepsilon_0(0))$. A similar argument holds for the case when it is in $\underline{\mathcal{X}_0}(0, \varepsilon_0(0))$.

Now we go back to (4.8) and rewrite it by partitioning the neighbor set $\mathcal{M}_i(k)$ of node i into two parts: The nodes which belong to $\overline{\mathcal{X}_0}(k, \varepsilon_0(k))$ and those that do not. Since node i has at least $F+1$ outside $\overline{\mathcal{X}_0}(k, \varepsilon_0(k))$, the latter set is nonempty. Hence, with positive probability, we obtain

$$\begin{aligned} x_i(k+1) = \Big\lfloor a_{ii}(k)x_i(k) &+ \sum_{j \in \mathcal{M}_i(k) \cap \overline{\mathcal{X}}_0} a_{ij}(k)x_j(k) \\ &+ \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}_0} a_{ij}(k)x_j(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k) \Big\rfloor, \end{aligned}$$

where we use the shorthand notation $\overline{\mathcal{X}}_0$ for $\overline{\mathcal{X}}_0(k, \varepsilon_0(k))$. Then, we can bound this from above as

$$\begin{aligned} x_i(k+1) &\leq \Bigg\lfloor a_{ii}(k)\overline{x}(k) + \sum_{j \in \mathcal{M}_i(k) \cap \overline{\mathcal{X}}_0} a_{ij}(k)\overline{x}(k) \\ &\quad + \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}_0} a_{ij}(k) \left(\overline{x}_0(k) - \varepsilon_0(k)\right) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k) \Bigg\rfloor \\ &= \Bigg\lfloor \left(1 - \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}_0} a_{ij}(k)\right)\overline{x}(k) \\ &\quad + \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}_0} a_{ij}(k) \left(\overline{x}_0(k) - \varepsilon_0(k)\right) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k) \Bigg\rfloor. \end{aligned} \tag{4.17}$$

We next show by induction that $\overline{x}(k) \leq \overline{x}_0(k) + \sigma_0$ with positive probability (and similarly, $\underline{x}(k) \geq \underline{x}_0(k) - \sigma_0$). For $k = 0$, by definition, we have $\overline{x}(0) = \overline{x}_0(0) + \sigma_0$. Suppose that $\overline{x}(k) \leq \overline{x}_0(k) + \sigma_0$ with positive probability. Then, from (4.9) and (4.10), with positive probability, we have

$$\begin{aligned} \overline{x}\,(k+1) &\leq \left\lfloor \overline{x}(k) + c_0 + c_1 \mathrm{e}^{-\alpha k} \right\rfloor \\ &\leq \left\lfloor (\overline{x}_0(k) + \sigma_0) + c_0 + c_1 \mathrm{e}^{-\alpha k} \right\rfloor \\ &= \lfloor \overline{x}_0(k+1) + \sigma_0 \rfloor \leq \overline{x}_0(k+1) + \sigma_0. \end{aligned}$$

Thus we have $\overline{x}(k) \leq \overline{x}_0(k) + \sigma_0$ with positive probability.

On the other hand, (4.17) can be further bounded as

$$\begin{aligned} x_i(k+1) &\leq \Big\lfloor \left(1 - \sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathcal{X}}_0} a_{ij}(k)\right)(\overline{x}_0(k)+\sigma_0) \\ &\quad + \sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathcal{X}}_0} a_{ij}(k)\left(\overline{x}_0(k)-\varepsilon_0(k)\right) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\hat{e}_j(k)\Big\rfloor \\ &\leq \Big\lfloor \overline{x}_0(k) + \sum_{j\in\mathcal{M}_i(k)\cap\overline{\mathcal{X}}_0} a_{ij}(k)\sigma_0 \\ &\quad - \sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathcal{X}}_0} a_{ij}(k)\varepsilon_0(k) + \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)|\hat{e}_j(k)|\Big\rfloor. \end{aligned} \tag{4.18}$$

We also show that from (4.12), it holds $\varepsilon_0(k) > 0$ for $k = 0, 1, \ldots, |\mathcal{R}|$. It is clear that $\varepsilon_0(k+1) < \varepsilon_0(k)$, and hence we need to guarantee only $\varepsilon_0(|\mathcal{R}|) > 0$. By (4.12), $\varepsilon_0(|\mathcal{R}|)$ can be written as

$$\begin{aligned} \varepsilon_0(|\mathcal{R}|) &= \gamma^{|\mathcal{R}|}\varepsilon_0(0) - \sum_{i=0}^{|\mathcal{R}|-1}\gamma^i(1-\gamma)\sigma_0 \\ &= \left(\gamma^{|\mathcal{R}|}\varepsilon - (1-\gamma^{|\mathcal{R}|})\sigma\right)V(0). \end{aligned}$$

This is positive because we have chosen σ as in (4.13).

Hence, (4.18) can be written as

$$\begin{aligned} x_i(k+1) &\leq \left\lfloor \overline{x}_0(k) + (1-\gamma)\,\sigma_0 - \gamma\varepsilon_0(k) + c_0 + c_1\mathrm{e}^{-\alpha k}\right\rfloor \\ &= \lfloor \overline{x}_0(k+1) - \varepsilon_0(k+1)\rfloor \\ &\leq \overline{x}_0(k+1) - \varepsilon_0(k+1), \end{aligned} \tag{4.19}$$

for $k = 0, 1, \ldots, |\mathcal{R}| - 1$, where in the first inequality, we used the fact that there always exists j not in $\overline{\mathcal{X}}_0\left(k, \varepsilon_0\left(k\right)\right)$ and the equality follows from (4.10) and (4.12). This relation shows that once an update happens at node i, then this

node will move out of $\overline{\mathcal{X}}_0\left(k+1, \varepsilon_0\left(k+1\right)\right)$ with positive probability. We note that inequality (4.19) also holds for the regular nodes that are not inside $\overline{\mathcal{X}}_0\left(k, \varepsilon_0\left(k\right)\right)$. This means that the nodes outside will not move in $\overline{\mathcal{X}}_0\left(k+1, \varepsilon_0\left(k+1\right)\right)$ with positive probability. Similar results hold for the other set $\underline{\mathcal{X}}_0(k+1, \varepsilon_0(k+1))$.

Hence, after time $|\mathcal{R}|$, all the regular nodes will be out of at least one of the two sets $\overline{\mathcal{X}}_0\left(|\mathcal{R}|, \varepsilon_0\left(|\mathcal{R}|\right)\right)$ and $\underline{\mathcal{X}}_0\left(|\mathcal{R}|, \varepsilon_0\left(|\mathcal{R}|\right)\right)$ with positive probability. We suppose that $\overline{\mathcal{X}}_0\left(|\mathcal{R}|, \varepsilon_0\left(|\mathcal{R}|\right)\right) \cap \mathcal{R}$ is empty. When such an event occurs, it clearly follows that $\overline{x}\left(|\mathcal{R}|\right) \leq \overline{x}_0\left(|\mathcal{R}|\right) - \varepsilon_0\left(|\mathcal{R}|\right)$. From the definition of $V\left(k\right)$, with positive probability, we have

$$
\begin{aligned}
V(N) &= \overline{x}\left(|\mathcal{R}|\right) - \underline{x}\left(|\mathcal{R}|\right) \\
&\leq \left(\overline{x}_0\left(|\mathcal{R}|\right) - \varepsilon_0\left(|\mathcal{R}|\right)\right) - \left(\underline{x}_0\left(|\mathcal{R}|\right) - \sigma_0\right) \\
&= \overline{x}_0\left(0\right) - \underline{x}_0\left(0\right) + 2c_0|\mathcal{R}| + 2\sum_{i=0}^{|\mathcal{R}|-1} c_1 \mathrm{e}^{-\alpha i} - \varepsilon_0\left(|\mathcal{R}|\right) + \sigma_0 \\
&= \left(\overline{x}\left(0\right) - \sigma_0\right) - \left(\underline{x}\left(0\right) + \sigma_0\right) + 2c_0 N + 2c_1 \frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}} - \varepsilon_0\left(|\mathcal{R}|\right) + \sigma_0 \\
&= V\left(0\right) + 2c_0|\mathcal{R}| + 2c_1 \frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}} - \sigma V\left(0\right) - \left(\gamma^{|\mathcal{R}|}\varepsilon - \left(1-\gamma^{|\mathcal{R}|}\right)\sigma\right) V\left(0\right) \\
&= \left(1 - \gamma^{|\mathcal{R}|}\left(\varepsilon + \sigma\right)\right) V\left(0\right) + 2c_0 N + 2c_1 \frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}}.
\end{aligned}
$$

By $\varepsilon + \sigma = 1/2$ in (4.13), with positive probability, we have

$$
V\left(|\mathcal{R}|\right) \leq \left(1 - \frac{\gamma^{|\mathcal{R}|}}{2}\right) V\left(0\right) + 2c_0|\mathcal{R}| + 2c_1 \frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}}.
$$

If there are more updates by node i after time $k = |\mathcal{R}|$, this argument can be

extended further. Hence, with positive probability, for any $\ell \geq 1$, it holds

$$\begin{aligned} V(\ell|\mathcal{R}|) &\leq \left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right) V((\ell-1)|\mathcal{R}|) + 2c_0|\mathcal{R}| + 2c_1 \mathrm{e}^{-(\ell-1)\alpha|\mathcal{R}|}\frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}} \\ &\leq \left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{\ell} V(0) + \sum_{t=0}^{\ell-1}\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{\ell-1-t}\left(2c_0 N + 2c_1\mathrm{e}^{-(t-1)\alpha|\mathcal{R}|}\frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}}\right) \\ &\leq \left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{\ell} V(0) + 2c_0|\mathcal{R}|\frac{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{\ell}}{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)} \\ &\quad + 2c_1\frac{1-\mathrm{e}^{-\alpha|\mathcal{R}|}}{1-\mathrm{e}^{-\alpha}}\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{\ell}\frac{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{-\ell}\mathrm{e}^{-\alpha|\mathcal{R}|\ell}}{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)^{-\ell}\mathrm{e}^{-\alpha|\mathcal{R}|}}. \end{aligned} \tag{4.20}$$

In (4.20), because $0 < \gamma < 1$, it is clear that as $\ell \to \infty$, the upper bound on the far right-hand side converges to

$$2c_0|\mathcal{R}|\frac{1}{1-\left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\right)} = \frac{4c_0|\mathcal{R}|}{\gamma^{|\mathcal{R}|}}.$$

This indicates that for any $\delta > 0$, with positive probability, there exists a finite $L_0 > 0$ such that for $\ell \geq L_0$, it holds $V(\ell|\mathcal{R}|) \leq 4c_0|\mathcal{R}|/\gamma^{|\mathcal{R}|} + \delta$.

Similarly, we can analyze $V(\ell|\mathcal{R}| + t)$ for $t = 0, 1, \ldots, |\mathcal{R}| - 1$ and obtain a bound like (4.20), where the difference is that $V(0)$ is replaced with $V(t)$. Hence, we have that with positive probability, there exists a finite $L_t > 0$ such that for $\ell \geq L_t$

$$V(\ell|\mathcal{R}| + t) \leq \frac{4c_0|\mathcal{R}|}{\gamma^{|\mathcal{R}|}} + \delta. \tag{4.21}$$

Thus, with positive probability, we have that $|x_i(k) - x_j(k)| \leq 4c_0|\mathcal{R}|/\gamma^{|\mathcal{R}|} + \delta$ for all $k \geq \max_t L_t|\mathcal{R}|$ and $i, j \in \mathcal{R}$.

On the other hand, $\hat{V}(k)$ can be similarly bounded. By (4.7),

$$\hat{V}(k) \leq V(k) + 2(c_0 + c_1 \mathrm{e}^{-\alpha k}).$$

Hence, by (4.21), with positive probability, there exists $L' > 1$ such that for $\ell > L'$ and $t = 0, 1, \ldots, |\mathcal{R}| - 1$,

$$\hat{V}(\ell|\mathcal{R}| + t) \leq \frac{4c_0|\mathcal{R}|}{\gamma^{|\mathcal{R}|}} + 2c_0 + \delta. \tag{4.22}$$

Since $V(k)$ and $\hat{V}(k)$ take integer values, and we can arbitrarily choose δ, it is now clear from (4.21) and (4.22) that with positive probability

$$V(k), \hat{V}(k) \leq \left\lfloor 2c_0 \left(\frac{2|\mathcal{R}|}{\gamma^{|\mathcal{R}|}} + 1 \right) \right\rfloor, \quad \forall k > \max\{L_t, L'\}|\mathcal{R}|.$$

As the last step, it remains to show (C3) in Lemma 4.2.1. When all regular current values and communicated values are inside $\mathcal{C}_\beta$ at time k', from (4.5), it is straightforward that all the regular values $x_i(k)$ cannot move outside $\mathcal{C}_\beta$ at time $k > k'$. Moreover, the same holds for the regular auxiliary values $\hat{x}_i(k)$ since they can take only the same values as $x_i(k)$ over time. Thus, (C3) has been established. This concludes the proof. ■

Theorem 4.2.1 deals with the quantized version of the event-based resilient consensus problem studied in [85]. It is interesting to note that for quantized consensus even for the case without malicious nodes, event-based schemes have not been considered much in the literature (see, e.g., [90]).

This protocol has several interesting features as follows.

(i) The $(F+1, F+1)$-robust graph is a necessary and sufficient condition for the conventional resilient consensus problems without event-triggered protocol for real-valued nodes in [45] and for quantized-valued nodes in [18].

(ii) Consensus can be achieved in finite time with some approximation due to the event-based scheme. Clearly, communication will be less with greater error in consensus at the end. We note that finite time convergence can be realized by algorithms for resilient quantized consensus in [18] without event-based rules though communication among agents is necessary at every step in the synchronous case.

(iii) In the triggering function, the parameter c_0 sets the level of approximation. As seen in (4.4), by using smaller c_0, the approximate level expressed by β becomes smaller. Observe that as $c_0 \to 0$, exact consensus with $\beta = 0$ becomes possible. We however show below that for exact consensus, it is in fact sufficient to use $c_0 < 1$. It is also noted that in the triggering function (4.1), the time-varying part $c_1 \mathrm{e}^{-\alpha(k+1)}$ affects only the transients. It can be replaced by any nonincreasing function of time that always takes nonnegative values and goes to zero in a finite number of steps. The analysis will remain essentially the same.

(iv) As a technical difference from the real-valued case studied in [85], the bound on the approximate consensus set $\mathcal{C}_\beta$ requires an arbitrarily small $\delta > 0$ in the derivation in the proof. This is necessary due to the probabilistic arguments there to guarantee that with positive probability the agents' values will eventually go into $\mathcal{C}_\beta$. With $\delta = 0$, the probability will be zero as in (4.20), it requires an infinite number of steps.

We now consider when exact consensus with $\mathcal{C}_0$ can be attained. It turns out that with the parameter $c_0 < 1$, this is possible. In this case, with a sufficiently large k_f, we have $c_0 + c_1 \mathrm{e}^{-\alpha k} < 1$ for $k > k_f$. This indicates that after time k_f, at each moment k the current value changes for any regular agent, an event is guaranteed to happen, and thus it holds $x_i(k) = \hat{x}_i(k)$. Thus, we can rewrite the

update rule (4.2) as

$$x_i(k+1) = x_i(k) + Q\left(\sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k) - x_i(k)\right)\right) \tag{4.23}$$

for $k > k_f$. This coincides with the quantized resilient consensus update rule of [18]. We state this fact as a corollary.

Corollary 4.2.1. *If $c_0 < 1$, then under the F-total model, the regular agents with the QE-MSR using* (4.2) *and* (4.3) *reach quantized resilient consensus with $\beta = 0$ in finite time almost surely if and only if the underlying graph is $(F+1, F+1)$-robust. The safety interval is given by $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$.*

We note that if we set $c_1 = 0$ together with $0 \leq c_0 < 1$, then the protocol will reduce to that in [18] from the initial time. However, it is one of the features of the quantized case that exact consensus can be guaranteed even if $c_0 > 0$. This is in contrast to the real-valued case studied in, e.g., [75], where the approximation error will always remain and its level depends on the size of $c_0 > 0$.

4.3 An alternative QE-MSR algorithm

We next provide our second resilient consensus algorithm, which will be referred to as Protocol 2. It is quite similar to the first protocol, but the system behavior as well as the approximate consensus bound are different.

The difference in the protocols is simple. For Protocol 2, to compute the new state $x_i(k+1)$ of agent i, we propose to replace in (4.3) the current state $x_i(k)$ with the last communicated state $\hat{x}_i(k)$. Notice that in Protocol 1, the data of its own comes in only through $x_i(k)$ and $\hat{x}_i(k)$ is not used. This may seem desirable since the current state $x_i(k)$ is newer, which might potentially help the convergence speed. An advantage of the approach for Protocol 2 is that $x_i(k+1)$

need not be stored but will be used only for checking the triggering condition. This structure in the protocol is motivated by those in [33] and [88].

Protocol 2 follows the steps of Protocol 1 except that in the local update of Step 3, for $k \in \mathbb{Z}_+$, every regular node $i \in \mathcal{R}$ updates its current state by

$$x_i(k+1) = Q\left(\hat{x}_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(\hat{x}_j(k) - \hat{x}_i(k)\right)\right). \tag{4.24}$$

As in Protocol 1, the initial regular communicated values satisfy $\hat{x}_i(0) \in \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$, $i \in \mathcal{R}$.

The results for Protocol 2 are presented in a manner parallel to those for Protocol 1. We first give the result for general case.

Theorem 4.3.1. *Under the F-total model, the regular agents with the QE-MSR using* (4.24) *and* (4.3) *reach bounded quantized resilient consensus in finite time almost surely if and only if the underlying graph is $(F+1, F+1)$-robust. The safety interval is given by $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$, and the approximate consensus set $\mathcal{C}_\beta$ is given with*

$$\beta = \min\left\{|\mathcal{S}|, \left\lfloor c_0 \frac{1-\gamma^{|\mathcal{R}|-1}}{\gamma^{|\mathcal{R}|-1}(1-\gamma)} \right\rfloor\right\}. \tag{4.25}$$

Outline of the Proof: The theorem can be proven by following similar lines as those in the proof of Theorem 4.2.1. For the sufficiency part, we must establish the three conditions (C1)–(C3) in Lemma 4.2.1. This can be done by following the analysis for Protocol 2 in the real-valued case from Chapter 3. For (C1) and (C3), we must show that $\overline{x}(k)$ and $\overline{\hat{x}}(k)$ are nonincreasing functions and that $\underline{x}(k)$ and $\underline{\hat{x}}(k)$ are nondecreasing functions of time k. For proving (C2), due to the use of the probabilistic quantizer in the current setting, the proof must be modified from the real-valued case in Chapter 3, where the protocol is deterministic. For

Protocol 2, certain parts must be shown to hold only with positive probability, which is enough to guarantee the condition (C2). ■

Compared with Protocol 1, Protocol 2 is simpler as only the data communicated via event-based protocols is used to achieve resilient consensus. In general, this feature results in slower triggering of transmission events and hence lower communication rate, especially when the values of the regular nodes become close to each other. As a consequence, larger error among regular nodes may remain for consensus when same parameters are used for the triggering functions in the two protocols. Interestingly, however, from the theoretical viewpoint, the advantage of Protocol 2 is that the upper bound in Theorem 4.3.1 for the approximate consensus set $\mathcal{C}_\beta$ is smaller than the one in Theorem 4.2.1. It is noted that the bound in Theorem 4.2.1 can be shown to be valid for the current case also by employing the proof method there.

From Theorem 4.3.1, we also observe that exact consensus can be attained by taking the parameter c_0 in the triggering rule small. Similarly to Protocol 1, it is in fact sufficient to set $c_0 < 1$, as demonstrated in the next corollary.

Corollary 4.3.1. *If $c_0 < 1$, then under the F-total model, the regular agents with the QE-MSR using* (4.24) *and* (4.3) *reach quantized resilient consensus with $\beta = 0$ if and only if the underlying graph is $(F+1, F+1)$-robust. The safety interval is given by $\mathcal{S} = \left[\underline{\hat{x}}(0), \overline{\hat{x}}(0)\right]$.*

Proof. The proof is similar to that of Corollary 4.2.1. After finite time steps in this protocol, the two states in each agent i become the same, $x_i(k) = \hat{x}_i(k)$. Thus, the update rule in (4.24) reduces to that of (4.23). □

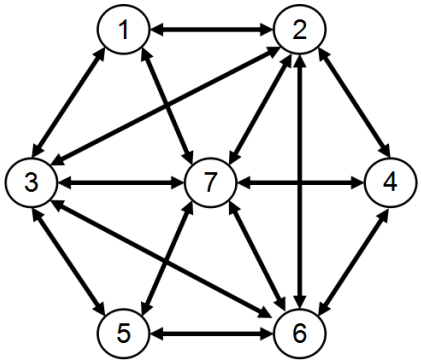


Figure 4.2: Network topology with $(2, 4)$-robustness

4.4 Numerical example

In this section, we illustrate the proposed two resilient consensus approaches via a numerical example.

We consider the multi-agent system with 7 nodes whose connectivity graph is shown in Fig. 4.2. As mentioned earlier, this graph is $(2, 4)$-robust. Hence, according to our theoretical results, with up to one malicious agent (i.e., $F = 1$), the regular agents should reach consensus. The parameters of the triggering function are chosen as $c_0 = 0.1$, $c_1 = 1$, and $\alpha = 2$. The initial states were set as $x(0) = [10\ 25\ 13\ 8\ 20\ 18\ 13]^T$ for all simulations.

Node 7 is chosen as the malicious node and continuously oscillates its value over time. The time responses for the two protocols are depicted in Figs. 4.3 and 4.4. In each plot, the time instants when each node broadcasts are shown by the markers $*$ in the color corresponding to that of its time response curve. As expected, we observe that both protocols reach exact resilient quantized consensus. Moreover, for this simulation, Protocol 2 is slower than Protocol 1 in convergence time.

Another simulation was performed by modifying the critical parameter c_0 in the triggering function as $c_0 = 1.1$. The time responses for the two protocols are

exhibited in Figs. 4.5 and 4.6. In this case, as expected, exact resilient quantized consensus cannot be guaranteed and approximation errors remain for both protocols. The error is larger for Protocol 2 though the number of transmissions required overall is smaller. Hence, we should highlight the tradeoff between the achievable level of consensus and the required communication among the agents.

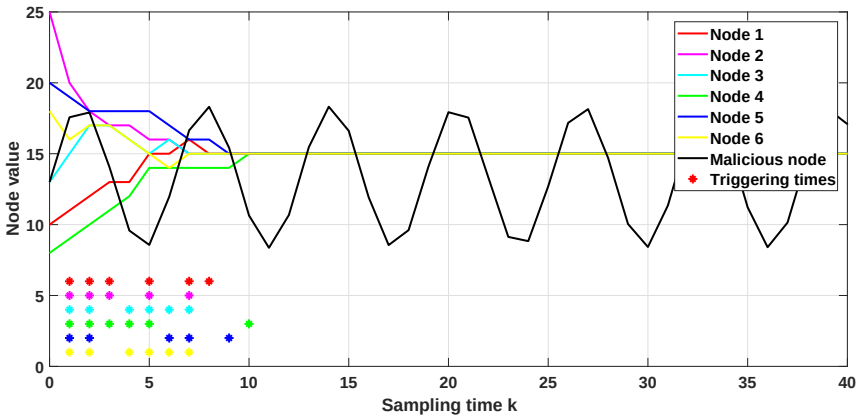


Figure 4.3: Protocol 1 with $c_0 = 0.1$

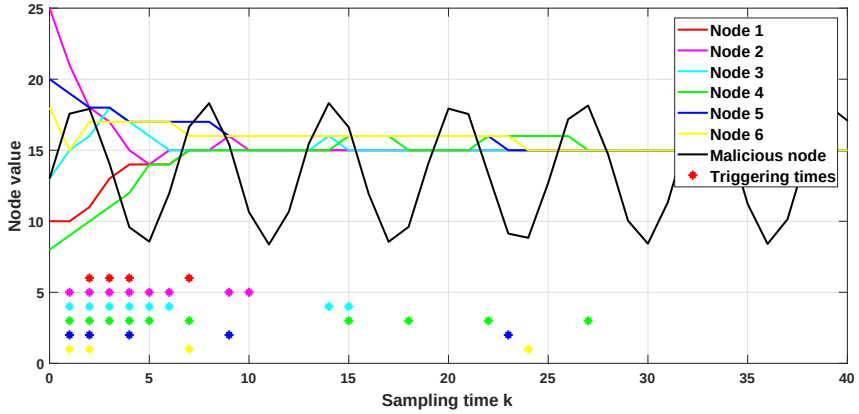


Figure 4.4: Protocol 2 with $c_0 = 0.1$

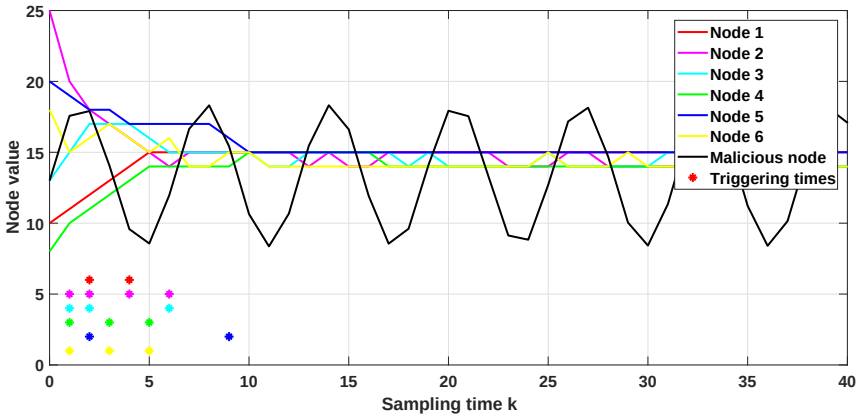


Figure 4.5: Protocol 1 with $c_0 = 1.1$

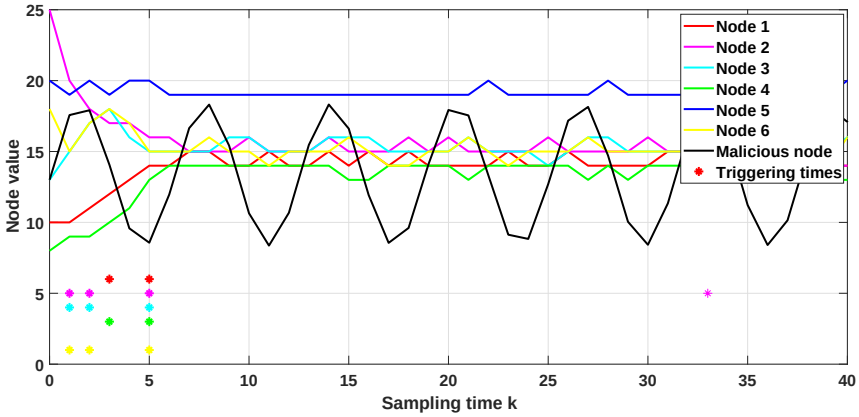


Figure 4.6: Protocol 2 with $c_0 = 1.1$

Chapter 5

A Distributed Model Predictive Scheme for Resilient Consensus with Input Constraints

In this chapter, we study the problem of resilient consensus in multi-agent networks with bounded input constraints. The resilient update rules takes account of the presence of attacks by malicious agents in the network. Each regular agent solves a constrained finite-time optimal problem with the states of its neighbors and updates its state based on a predetermined update rule. Schemes are proposed to solve the problem with synchronous and asynchronous communications, assuming that the maximum number of malicious nodes is known. We derive algorithms which ignore the large and small values from neighbors to avoid the influence of the malicious nodes. It is guaranteed to attain resilient consensus under the topological condition expressed in terms of graph robustness. Simulation examples are provided to demonstrate the effectiveness of the proposed algorithm. This part is published in [86].

5.1 Problem formulation

5.1.1 Model predictive consensus protocol with input constraints

Consider the multi-agent system where each agent is described by the following discrete-time single-integrator model

$$x_i(k+1) = x_i(k) + u_i(k), \tag{5.1}$$

where $x_i(k) \in \mathbb{R}$ is its state and $u_i(k) \in \mathbb{R}$ is the control input.

The basic objective of these agents is to attain consensus, that is, $x_i(k) - x_j(k) \to 0$ as $k \to \infty$ for all $i, j \in \mathcal{V}$ in an iterative manner by interacting with each other through exchanges of their state values. Under the model predictive control scheme, the control input of agent i is determined by locally solving an optimal control problem with a finite horizon formulated as follows. Let $N_i \geq 1$ be the length of the prediction horizon for node i. We introduce the input sequence $U_i(k) = [u_i(k) \quad u_i(k+1) \quad \cdots \quad u_i(k+N_i-1)]^T$ and the cost function is set as

$$J_i(x_i(k), z_i(k), U_i(k)) = J_i^x(x_i(k), z_i(k), U_i(k)) + J_i^u(U_i(k)),$$

where the costs for the states and the inputs are given, respectively, as

$$\begin{aligned} J_i^x(x_i(k), z_i(k), U_i(k)) &= \alpha_i \sum_{j=1}^{N_i} |x_i(k+j) - z_i(k)|^2, \\ J_i^u(U_i(k)) &= \beta_i \sum_{j=1}^{N_i-1} |u_i(k+j)|^2. \end{aligned} \tag{5.2}$$

Here, $z_i(k)$ is called the target point for agent i and is the state that the agent aims at reaching in the next step; it will be discussed more later. The parameters

α_i and β_i are the weights.

Agent i calculates the optimal control sequence the optimal control sequence $U_i^o(k)$ by solving the following problem at each time k:

$$\min_{U_i(k)} J_i\left(x_i\left(k\right), z_i\left(k\right), U_i\left(k\right)\right), \tag{5.3}$$

subject to the agent dynamics (5.1) and the input constraint given by

$$\left|u_i\left(k\right)\right| \leq u_{i,\max}. \tag{5.4}$$

We write the optimal solution $U_i^o(k)$ as

$$U_i^o\left(k\right) = \left[u_i^o\left(k|k\right), \cdots, u_i^o\left(k+N_i-1|k\right)\right]^T,$$

and choose our control law as $u_i(k) = u_i^o(k|k)$ for $k \in \mathbb{Z}_+$. In what follows, we may simplify the notation and write $u_i^o(k)$ for $u_i^o(k|k)$ whenever no confusion arises.

5.1.2 MP-MSR algorithm

Here, we outline the fault-tolerant consensus protocol to solve the resilient consensus problem with input constraint. The algorithm is called the Model Predictive based Mean Subsequence Reduced (MP-MSR) algorithm. We consider the adversary model of $F - total$ malicious.

The MP-MSR has four steps as shown below:

Algorithm 5.1.1. *(MP-MSR Algorithm, Protocol 1)*

1. (Collecting neighbour information) *At time step $k \in \mathbb{Z}_+$, every regular node i collects the neighbours' value $x_j(k)$, $j \in \mathcal{N}_i$, and sorts them from the largest to the smallest (including its own value $x_i(k)$).*

2. (Deleting suspicious values) *Comparing with $x_i(k)$, node i removes the F largest and F smallest values from its neighbours. If the number of values larger (or smaller) than $x_i(k)$ is less than F, then delete all of them. The deleted data is considered as suspicious data and will not be used in the following local updates. The set of the remaining values is written by $\mathcal{M}_i(k) \subset \mathcal{N}_i$.*

3. (Local optimization) *Every regular node i solves the optimization problem*

$$\min_{U_i(k)} J_i\left(x_i\left(k\right), z_i\left(k\right), U_i\left(k\right)\right)$$

subject to the constraints (5.1) *and* (5.4)*, where the target point $z_i(k)$ is given by*

$$z_i\left(k\right) = x_i\left(k\right) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}\left(k\right)\left(x_j\left(k\right) - x_i\left(k\right)\right). \tag{5.5}$$

4. (Local update) *The initial input is set as $u_i(k) = u_i^o(k|k)$, then the state is updated by* (5.1).

It will be established that as long as the agent network is sufficiently connected in the sense of robustness, the MP-MSR algorithm is capable to reach resilient consensus.

5.2 Main results on synchronous MP-MSR algorithm

In this section, the main results on the synchronous MSR algorithm based on model predictive control are presented.

5.2.1 Properties on the optimal control input

Here, we provide a preliminary result concerning the optimal control input. We first introduce a modified expression of the update rule (5.1) for the regular nodes under the model predictive control scheme discussed above. It takes the form of conventional consensus but with a time-varying parameter. Specifically, it is written as

$$x_i(k+1) = x_i(k) + \eta_i(k) \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k) - x_i(k)\right), \tag{5.6}$$

where the parameter $\eta_i(k)$ is given by

$$\eta_i(k) = \begin{cases} \frac{u_i^o(k|k)}{\sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)(x_j(k) - x_i(k))} & \text{if } \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k) - x_i(k)\right) \neq 0, \\ 1 \qquad \text{otherwise.} \end{cases} \tag{5.7}$$

Regarding the parameter $\eta_i(k)$, the following result exhibits an important property, showing that it is lower bounded by a constant at all times.

Proposition 5.2.1. *Under the MP-MSR algorithm, there exists a positive constant $c \in (0,1)$ such that $c \leq \eta_i \leq 1$ for each regular node i and each $k \in \mathbb{Z}_+$.*

The proof of this proposition uses certain geometric arguments, for which we need to introduce the notion of paths in the state space. Given the control sequence $U_i(k)$ of length N_i, we call the sequence of states $X_i(x_i(k), U_i(k)) = [x_i(k), x_i(k+1), \cdots, x_i(k+N_i)]$ to be the N_i-path from $x_i(k)$ to $x_i(k+N_i)$. More specifically, such paths are defined as below.

Definition 5.2.1. *The N-path $T_A = (x_0, \cdots, x_{N-1})$ is pointing towards $z \in \mathbb{R}$ if*

1. $|x_{j+1} - z| \leq |x_j - z|$, $j \in \{0, 1, \ldots, N-2\}$,

2. $x_j \in [\min(x_1, z), \max(x_1, z)]$, $j \in \{1, 2, \ldots, N\}$.

The following lemma from Theorem 5 in [23] gives the necessary geometric properties to be used in our analysis.

Lemma 5.2.1. *Let* $[x_0 \quad \cdots \quad x_{N-1}]^T \in \mathbb{R}^N$ *be an N-path. Given* $z \in \mathbb{R}$*, there always exists an N-path* $[y_0 \quad \cdots \quad y_{N-1}]^T \in \mathbb{R}^N$ *with* $y_0 = x_0$ *pointing towards* z *and satisfying the following inequalities:*

1. $|y_j - z| \leq |x_j - z|$, $j \in \{0, 1, \ldots, N-1\}$,

2. $|y_{j+1} - y_j| \leq |x_{j+1} - x_j|$, $j \in \{0, 1, \ldots, N-2\}$.

This lemma gives the basic properties of N-path, these two properties can guarantee an shortest path from one point to another. We next present basic properties of the optimal control input.

Lemma 5.2.2. *Under the MP-MSR algorithm, the following facts hold for each regular node i:*

1) $X_i^o\left(x_i\left(k\right), U_i^o\left(k\right)\right)$ *is an* N_i*-path from* $x_i(k)$ *to* $z_i(k)$.

2) If $x_i(k) \neq z_i(k)$*, then* $U_i^o(k) \neq 0$.

3) If $x_i(k) \neq z_i(k)$*, then* $u_i^o(k|k) \neq 0$.

Proof. 1). We prove by contradiction. Suppose that $X_i^o\left(x_i\left(k\right), U_i^o\left(k\right)\right)$ is not pointing towards $z_i(k)$. From Lemma 5.2.1, we know that there exists an N_i-path $\hat{X}_i\left(k\right) = \left[\hat{x}_i\left(k\right), \cdots, \hat{x}_i\left(k+N_i\right)\right]$ with $\hat{x}_i\left(k\right) = x_i^o\left(k\right)$ pointing towards $z_i(k)$. Then, it follows that for $\forall j = 0, \ldots, N_i - 1$,

$$\left|\hat{x}_j\left(k+j+1\right) - z_i\left(k\right)\right| \leq \left|x_i^o\left(k+j+1\right) - z_i\left(k\right)\right|,$$
$$\left|\hat{x}_j\left(k+j+1\right) - \hat{x}_j\left(k+j\right)\right| \leq \left|x_i^o\left(k+j+1\right) - x_i^o\left(k+j\right)\right|.$$

From the definition of $J_i^x\left(x_i\left(k\right), z_i\left(k\right), U_i\left(k\right)\right)$ in (5.2), we have the following inequalities:

$$J_i^x\left(\hat{x}_i\left(k\right), \hat{z}_i\left(k\right), \hat{U}_i\left(k\right)\right) \leq J_i^x\left(x_i^o\left(k\right), z_i^o\left(k\right), U_i^o\left(k\right)\right),$$
$$J_i^u\left(\hat{x}_i\left(k\right), \hat{z}_i\left(k\right), \hat{U}_i\left(k\right)\right) \leq J_i^u\left(x_i^o\left(k\right), z_i^o\left(k\right), U_i^o\left(k\right)\right).$$

This implies that $X_i^o(x_i(k), U_i^o(k))$ is not optimal. Hence the optimal path $X_i^o(x_i(k), U_i^o(k))$ has to point towards $z_i(k)$.

2). We prove again by contradiction. Suppose that $x_i(k) \neq z_i(k)$, and the optimal solution is $U_i^o(k) = 0$. Then,

$$J_i\left(x_i\left(k\right), z_i\left(k\right), U_i^o\left(k\right)\right) = \alpha_i N_i |x_i\left(k\right) - z_i\left(k\right)|^2. \tag{5.8}$$

Take the control vector as $\overline{U}_i\left(k\right) = \left[\begin{array}{cccc} \delta\left(z_i\left(k\right) - x_i\left(k\right)\right) & 0 & \cdots & 0 \end{array}\right]^T$, with $0 < \delta < 1$ and $\left|\delta\left(z_i\left(k\right) - x_i\left(k\right)\right)\right| \leq u_{i,\max}$. The path resulting from this control is denoted by $\overline{X}_i\left(lN_i\right) = \left[\overline{x}_i\left(k\right), \cdots, \overline{x}_i\left(k + N_i\right)\right]$. Then, we have

$$J_i^u\left(x_i\left(k\right), z_i\left(lN_i\right), \overline{U}_i\left(k\right)\right) = \beta_i \delta^2 |x_i\left(k\right) - z_i\left(lN_i\right)|^2. \tag{5.9}$$

By this control, we can obtain the states as $\overline{x}_i\left(k + N_i\right) = \overline{x}_i\left(k + N_i - 1\right) = \cdots = \overline{x}_i\left(k + 1\right)$ and

$$\begin{aligned} x_i\left(k+1\right) &= x_i\left(k\right) + u_i\left(k\right) = x_i\left(k\right) + \delta\left(z_i\left(k\right) - x_i\left(k\right)\right) \\ &= \left(1 - \delta\right) x_i\left(k\right) + \delta z_i\left(k\right). \end{aligned}$$

Then, we obtain

$$J_i^x\left(x_i\left(k\right), z_i\left(k\right), \overline{U}_i\left(k\right)\right) = \alpha_i N_i |x_i\left(k+1\right) - z_i\left(lN_i\right)|^2$$
$$= \alpha_i N_i (1-\delta)^2 |x_i\left(k\right) - z_i\left(lN_i\right)|^2. \tag{5.10}$$

By (5.9) and (5.10), the cost function is

$$J_i\left(x_i\left(k\right), z_i\left(k\right), \overline{U}_i\left(k\right)\right) = \left(\alpha_i N_i (1-\delta)^2 + \beta_i \delta^2\right) |x_i\left(k\right) - z_i\left(k\right)|^2. \tag{5.11}$$

By comparing (5.8) and (5.11), we choose δ sufficiently small that

$$\delta < \frac{2\alpha_i N_i}{\alpha_i N_i + \beta_i}.$$

It will then follow

$$J_i\left(x_i\left(k\right), z_i\left(k\right), \overline{U}_i\left(k\right)\right) < J_i\left(x_i\left(k\right), z_i\left(k\right), U_i^o\left(k\right)\right).$$

Hence, $U_i^o(k) = 0$ is not the optimal solution.

3). This fact is shown by contradiction as well. Suppose that the optimal solution takes the form as

$$U_i^o(k) = \left[\begin{array}{cccc} 0 & u_i^o\left(k+1\right) & \cdots & u_i^o\left(k+N_i-1\right) \end{array}\right]^T.$$

We also consider the input

$$\overline{U}_i\left(k\right) = \left[\begin{array}{cccc} u_i^o\left(k+1\right) & \cdots & u_i^o\left(k+N_i-1\right) & 0 \end{array}\right]^T.$$

Denote the state resulting from this input by $\overline{x}_i(k+1), \ldots, \overline{x}_i(k+N_i)$ with $\overline{x}_i(k) = x_i(k)$. It is easy to see that this control input satisfies the input constraint (5.4)

just as the optimal solution does. Then it follows

$$J_i^u\left(x_i\left(k\right),z_i\left(k\right),\overline{U}_i\left(k\right)\right)=J_i^u\left(x_i\left(k\right),z_i\left(k\right),U_i^o\left(k\right)\right),$$

and

$$\begin{aligned}&J_i\left(x_i\left(k\right),z_i\left(k\right),\overline{U}_i\left(k\right)\right)-J_i\left(x_i\left(k\right),z_i\left(k\right),U_i^o\left(k\right)\right)\\&\quad=\alpha_i\sum_{j=1}^{N_i}\left|\overline{x}_i\left(k+j\right)-z_i\left(k\right)\right|-\alpha_i\sum_{j=1}^{N_i}\left|x_i^o\left(k+j\right)-z_i\left(k\right)\right|\\&\quad=\alpha_i\Big|x_i\left(k\right)+\sum_{j=1}^{N_i-1}u_i^o\left(k+j\right)-z_i\left(k\right)\Big|-\alpha_i\left|x_i\left(k\right)-z_i\left(k\right)\right|.\end{aligned}$$

From 1) and 2), we know that compared with $x_i(k)$, $x_i\left(k\right)+\sum_{j=1}^{N_i-1}u_i^o\left(k+j\right)$ is closer to $z_i(k)$. Hence,

$$\alpha_i\Big|x_i\left(k\right)+\sum_{j=1}^{N_i-1}u_i^o\left(k+j\right)-z_i\left(k\right)\Big|-\alpha_i\left|x_i\left(k\right)-z_i\left(k\right)\right|<0.$$

This indicates that $U_i^o(k)$ is not the optimal solution. Therefore, we have $u_i^o(k|k)\neq 0$ if $x_i(k)\neq z_i(k)$. □

Proof of Proposition 5.2.1: It is first proven that

$$\left|u_i^o(k|k)\right|\leq\left|\sum_{j\in\mathcal{M}_i(k)}a_{ij}\left(k\right)\left(x_j\left(k\right)-x_i\left(k\right)\right)\right| \tag{5.12}$$

if the right-hand side is nonzero. This will be shown by contradiction. Suppose that

$$\left|u_i^o(k|k)\right|>\left|\sum_{j\in\mathcal{M}_i(k)}a_{ij}\left(k\right)\left(x_j\left(k\right)-x_i\left(k\right)\right)\right|,$$

and

$$U_i^o(k) = \left[\begin{array}{cccc} u_i^o(k) & u_i^o(k+1) & \cdots & u_i^o(k+N_i-1) \end{array} \right]^T.$$

Then, we choose a different control input as

$$\overline{u}_i(k) = \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) \left(x_j(k) - x_i(k) \right),$$

and let

$$\overline{U}_i(k) = \left[\begin{array}{cccc} \overline{u}_i(k) & u_i^o(k+1) & \cdots & u_i^o(k+N_i-1) \end{array} \right]^T.$$

It is obvious that we are led to the contradiction:

$$J_i^x\left(x_i(k), z_i(k), U_i^o(k)\right) > J_i^x\left(x_i(k), z_i(k), \overline{U}_i(k)\right),$$
$$J_i^u\left(x_i(k), z_i(k), U_i^o(k)\right) > J_i^u\left(x_i(k), z_i(k), \overline{U}_i(k)\right).$$

Therefore, (5.12) has been shown.

Next, we prove that the signs of $u_i^o(k)$ and $\sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k) - x_i(k)\right)$ are the same. To show by contradiction, suppose that $\sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k) - x_i(k)\right)$ is positive, and $u_i^o(k)$ is negative (since $u_i^o \neq 0$ from property 3) of Lemma 5.2.2 where the optimal solution is $U_i^o(k) = \left[\begin{array}{cccc} u_i^o(k) & u_i^o(k+1) & \cdots & u_i^o(k+N_i-1) \end{array} \right]^T$. From the update rule (5.1), we know that

$$x_i(k+1) \notin \left[\min\left(x_i(k), z_i(k)\right), \max\left(x_i(k), z_i(k)\right)\right],$$

which is a contradiction with 1).

By applying (5.12) and the fact that the signs of $u_i^o(k)$ and $\sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k) - x_i(k)\right)$ are the same, we know that $0 < \eta_i(k) \leq 1$. It remains to show that there exists a positive constant $c \in (0,1)$ such that $c \leq \eta_i(k)$. We prove this fact by discussing the value of $u_i^o(k)$. Suppose that $u_i^o(k)$ and

$\sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k)-x_i(k)\right)$ are positive. In the following, the analysis is divided into two cases.

(i) The case of $u_{i,\max} \geq \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k)-x_i(k)\right)$: Assume $\beta_i \leq \alpha_i$, and let the optimal control sequence be

$$U_i^o(k) = \left[\begin{array}{cccc} u_i^o(k) & u_i^o(k+1) & \cdots & u_i^o(k+N_i-1) \end{array}\right]^T.$$

Then, we choose another control input as

$$\overline{u}_i(k) = \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k)-x_i(k)\right),$$

and the corresponding sequence as

$$\overline{U}_i(k) = \left[\begin{array}{cccc} \overline{u}_i(k) & 0 & \cdots & 0 \end{array}\right]^T.$$

It follows that

$$J_i\left(x_i(k), z_i(k), U_i^o(k)\right) > \alpha_i \left| u_i^o(k) - \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k)-x_i(k)\right)\right|^2,$$

and

$$J_i\left(x_i(k), z_i(k), \overline{U}_i(k)\right) = \beta_i \left| \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k)-x_i(k)\right)\right|^2.$$

Now, consider the case

$$u_i^o(k) \leq \left(1-\sqrt{\frac{\beta_i}{\alpha_i}}\right) \sum_{j\in\mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k)-x_i(k)\right).$$

We then arrive at $J_i\left(x_i\left(k\right), z_i\left(k\right), U_i^o\left(k\right)\right) > J_i\left(x_i\left(k\right), z_i\left(k\right), \overline{U}_i\left(k\right)\right)$, which is a contradiction. Hence, we can take $c_1 = 1 - \sqrt{\beta_i/\alpha_i}$ and have $\eta_i\left(k\right) > c_1$. In the case of $\beta_i \geq \alpha_i$, we can similarly find a constant c_2.

(ii) The case of $u_{i,\max} < \sum_{j\in\mathcal{M}_i(k)} a_{ij}\left(k\right)\left(x_j\left(k\right) - x_i\left(k\right)\right)$: Assume that $\beta_i \leq (1/2)\alpha_i$. Then, the corresponding optimal input becomes

$$U_i^o\left(k\right) = \left[\begin{array}{cccc} u_i^o\left(k\right) & u_i^o\left(k+1\right) & \cdots & u_i^o\left(k+N_i-1\right) \end{array}\right]^T.$$

We can choose another control input $\overline{u}_i(k) = u_{i,\max}$, where the corresponding sequence becomes

$$\overline{U}_i\left(k\right) = [\overline{u}_i\left(k\right) \quad -\overline{u}_i\left(k\right) + u_i^o\left(k\right) + u_i^o\left(k+1\right) \quad u_i^o\left(k+2\right) \quad \cdots \quad]^T.$$

It holds that

$$\begin{aligned} & J_i\left(x_i\left(k\right), z_i\left(k\right), U_i^o\left(k\right)\right) \\ & = \alpha_i \left| u_i^o\left(k\right) - \sum_{j\in\mathcal{M}_i(k)} a_{ij}\left(k\right)\left(x_j\left(k\right) - x_i\left(k\right)\right) \right|^2 \\ & \quad + \alpha_i \sum_{d=1}^{N_i-1} \left| u_i^o\left(k\right) + \sum_{e=1}^{d} u_i^o\left(k+e\right) - \sum_{j\in\mathcal{M}_i(k)} a_{ij}\left(k\right)\left(x_j\left(k\right) - x_i\left(k\right)\right) \right|^2 \\ & \quad + \beta_i |u_i^o\left(k\right)|^2 + \beta_i \sum_{d=1}^{N_i-1} |u_i^o\left(k+d\right)|^2, \end{aligned} \tag{5.13}$$

and

$$\begin{aligned}
&J_i\left(x_i\left(k\right), z_i\left(k\right), \overline{U}_i\left(k\right)\right) \\
&= \alpha_i \left| u_{i,\max} - \sum_{j\in\mathcal{M}_i(k)} a_{ij}\left(k\right)\left(x_j\left(k\right) - x_i\left(k\right)\right)\right|^2 \\
&\quad + \alpha_i \sum_{d=1}^{N_i-1} \left| u_i^o\left(k\right) + \sum_{e=1}^{d} u_i^o\left(k+e\right) - \sum_{j\in\mathcal{M}_i(k)} a_{ij}\left(k\right)\left(x_j\left(k\right) - x_i\left(k\right)\right)\right|^2 \\
&\quad + \beta_i |u_{i,\max}|^2 + \beta_i |-u_{i,\max} + u_i^o\left(k\right) + u_i^o\left(k+1\right)|^2 + \beta_i \sum_{d=2}^{N_i-1} |u_i^o\left(k+d\right)|^2.
\end{aligned} \tag{5.14}$$

By comparing (5.13) and (5.14), if

$$u_i^o\left(k\right) < 2\sum_{j\in\mathcal{M}_i(k)} a_{ij}\left(k\right)\left(x_j\left(k\right) - x_i\left(k\right)\right) - \left(1 + \frac{2\beta_i}{\alpha_i}\right) u_{i,\max},$$

then we arrive at

$$J_i\left(x_i\left(k\right), z_i\left(k\right), U_i^o\left(k\right)\right) > J_i\left(x_i\left(k\right), z_i\left(k\right), \overline{U}_i\left(k\right)\right),$$

which is a contradiction. So we have

$$\eta_i\left(k\right) \geq 2 - \left(1 + \frac{2\beta_i}{\alpha_i}\right) \frac{u_{i,\max}}{\sum_{j\in\mathcal{M}_i(k)} a_{ij}\left(k\right)\left(x_j\left(k\right) - x_i\left(k\right)\right)} \geq \frac{\alpha_i - 2\beta_i}{\alpha_i}.$$

Thus we can take $c_3 = (\alpha_i - 2\beta_i)/\alpha_i$ and have $\eta_i\left(k\right) \geq c_3$. We note that in case of $\beta_i \geq \frac{1}{2}\alpha_i$, we can similarly find such a constant c_4.

In conclusion, we can always find a constant $c = \min\{c_1, c_2, c_3, c_4\}$ such that $c \leq \eta_i(k) \leq 1$. ■

5.2.2 Resilient consensus via the MP-MSR algorithm

Based on the properties discussed in Proposition 5.2.1, we can proceed to obtain the main result of this chapter, stating that resilient consensus can be attained under the MP-MSR algorithm. To this end, we introduce the maxima and minima of the states of the regular agents: Let

$$\overline{x}\left(k\right)=\max_{i\in\mathcal{R}}x_i\left(k\right),\quad \underline{x}\left(k\right)=\min_{i\in\mathcal{R}}x_i\left(k\right)$$

The safety interval $\mathcal{S}$ is chosen as

$$\mathcal{S}=[\underline{x}\left(0\right),\overline{x}\left(0\right)].$$

Theorem 5.2.1. *Under the F-total malicious model, the normal agents with MP-MSR will reach resilient consensus if and only if the underlying graph is $(F+1,F+1)$-robust. The safety interval is given by $\mathcal{S}=[\underline{x}(0),\overline{x}(0)]$.*

Proof. (Necessity) The necessity part essentially follows from [45].

(Sufficiency) We first show the safety condition. At time $k=0$, by (5.5), we have

$$z_i\left(0\right)=\left(1-\sum_{j\in\mathcal{M}_i(k)}a_{ij}\left(0\right)\right)x_i\left(0\right)+\sum_{j\in\mathcal{M}_i(k)}a_{ij}\left(0\right)x_j\left(0\right).$$

It is easy to see that $\underline{x}(0)\leq x_i(0)\leq\overline{x}(0)$ and $\underline{x}(0)\leq z_i(0)\leq\overline{x}(0)$. We next assume that at time k, it holds $\underline{x}(0)\leq x_i(k)\leq\overline{x}(0)$ and $\underline{x}(0)\leq z_i(k)\leq\overline{x}(0)$. Again, from (5.5) we have

$$z_i\left(k+1\right)=x_i\left(k+1\right)+\sum_{j\in\mathcal{M}_i(k+1)}a_{ij}\left(k+1\right)\left(x_j\left(k+1\right)-x_i\left(k+1\right)\right).\qquad(5.15)$$

Based on the analysis in Proposition 5.2.1, we have

$$x_i\left(k+1\right) \in \left[\min\left(x_i\left(k\right), z_i\left(k\right)\right), \max\left(x_i\left(k\right), z_i\left(k\right)\right)\right]. \tag{5.16}$$

By applying $\underline{x}(0) \leq x_i(k) \leq \overline{x}(0)$ and $\underline{x}(0) \leq z_i(k) \leq \overline{x}(0)$ we have $\underline{x}(0) \leq x_i(k+1) \leq \overline{x}(0)$, which implies that $\underline{x}(0) \leq z_i(k+1) \leq \overline{x}(0)$ from (5.15).

Next we show the consensus condition part. Let $V(k) = \overline{x}(k) - \underline{x}(k)$ for $k \in \mathbb{Z}_+$. Here, we fix the time k. Let the sequence $\varepsilon_0(l), l \geq k$ be given by

$$\varepsilon_0(l) = \begin{cases} \frac{\overline{x}(k)-\underline{x}(k)}{2} & \text{if } l = k, \\ \eta_i(l-1)\gamma\varepsilon_0(l-1) & \text{if } l > k. \end{cases} \tag{5.17}$$

Further, introduce the two sets

$$\overline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right) = \left\{j \in \mathcal{V} : x_j\left(k'\right) > \overline{x}\left(k\right) - \varepsilon_0\left(k'\right)\right\},$$
$$\underline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right) = \left\{j \in \mathcal{V} : x_j\left(k'\right) < \underline{x}\left(k\right) + \varepsilon_0\left(k'\right)\right\},$$

where $k' \geq k$. Then, by the choice of $\varepsilon_0(k)$ in (5.17), it is clear that these two sets $\overline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right)$ and $\underline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right)$ with $k' = k$ are disjoint. By assumption, the graph is $(F+1, F+1)$-robust. Thus, by definition, one of the sets $\overline{\mathcal{X}}\left(k, k, \varepsilon_0\left(k\right)\right)$ and $\underline{\mathcal{X}}\left(k, k, \varepsilon_0\left(k\right)\right)$ has at least $F+1$ nodes having neighbours from outside the set to which it belongs. We first suppose that the former set $\overline{\mathcal{X}}\left(k, k, \varepsilon_0\left(k\right)\right)$ has this property. It further indicates that this set has one regular node i having neighbors outside.

We focus on this node i and divide its neighbor set $\mathcal{M}_i(k)$ into two parts: $\mathcal{M}_i(k) \cap \overline{\mathcal{X}}(k, k, \varepsilon_0(k))$ and $\mathcal{M}_i(k) \setminus \overline{\mathcal{X}}(k, k, \varepsilon_0(k))$; note that, by the choice of node i, the latter set is nonempty. We use the short-hand notation $\overline{\mathcal{X}}$ for $\overline{\mathcal{X}}(k, k, \varepsilon_0(k))$.

Then, from (5.6), it follows

$$\begin{aligned} x_i(k+1) &= \left(1-\eta_i(k)\sum_{j\in\mathcal{M}_i(k)}a_{ij}(k)\right)x_i(k)+\eta_i(k)\sum_{j\in\mathcal{M}(k)\cap\overline{\mathcal{X}}}a_{ij}(k)x_j(k) \\ &\quad +\eta_i(k)\sum_{j\in\mathcal{M}(k)\setminus\overline{\mathcal{X}}}a_{ij}(k)x_j(k) \\ &\leq \left(1-\eta_i(k)\sum_{j\in\mathcal{M}_i(k)}a_{ij}(k)\right)\overline{x}(k)+\eta_i(k)\sum_{j\in\mathcal{M}(k)\cap\overline{\mathcal{X}}}a_{ij}(k)\overline{x}(k) \\ &\quad +\eta_i(k)\sum_{j\in\mathcal{M}(k)\setminus\overline{\mathcal{X}}}a_{ij}(k)\left(\overline{x}(k)-\varepsilon_0(k)\right) \\ &= \overline{x}(k)-\eta_i(k)\sum_{j\in\mathcal{M}(k)\setminus\overline{\mathcal{X}}}a_{ij}(k)\varepsilon_0(k). \end{aligned} \tag{5.18}$$

Because node i has one or more neighbors outside $\overline{\mathcal{X}}(k,k,\varepsilon_0(k))$, it holds from (5.18)

$$x_i(k+1)\leq\overline{x}(k)-\eta_i(k)\gamma\varepsilon_0(k). \tag{5.19}$$

This inequality indicates that after the update, the regular node $i\in\overline{\mathcal{X}}(k,k,\varepsilon_0(k))$ moves out of $\overline{\mathcal{X}}(k,k+1,\varepsilon_0(k+1))$ due to (5.17).

Note that this inequality (5.18) also holds for all regular nodes outside $\overline{\mathcal{X}}(k,k,\varepsilon_0(k))$, and thus, they do not enter $\overline{\mathcal{X}}(k,k+1,\varepsilon_0(k+1))$. Hence, the number of regular nodes in the set $\overline{\mathcal{X}}(k,k+1,\varepsilon_0(k+1))$ decreases from that in $\overline{\mathcal{X}}(k,k,\varepsilon_0(k))$:

$$\left|\overline{\mathcal{X}}(k,k+1,\varepsilon_0(k+1))\cap\mathcal{R}\right|<\left|\overline{\mathcal{X}}(k,k,\varepsilon_0(k))\cap\mathcal{R}\right|.$$

The same argument holds if the set $\underline{\mathcal{X}}(k,k+1,\varepsilon_0(k+1))$ has the property that one of its nodes has a regular node as a neighbor outside the set. Thus, we can

repeat this argument for $|\mathcal{R}|$ steps, at which point, the set

$$\overline{\mathcal{X}}\left(k, k+|\mathcal{R}|, \varepsilon_0\left(k+|\mathcal{R}|\right)\right) \tag{5.20}$$

or

$$\underline{\mathcal{X}}\left(k, k+|\mathcal{R}|, \varepsilon_0\left(k+|\mathcal{R}|\right)\right) \tag{5.21}$$

will become empty.

We first consider the case where the set in (5.20) is empty. Then, we have

$$x_i\left(k+|\mathcal{R}|\right) \leq \overline{x}\left(k\right) - \left(\prod_{j=0}^{|\mathcal{R}|-1} \eta_i\left(k+j\right)\right) \gamma^{|\mathcal{R}|} \varepsilon_0\left(k\right), \forall i \in \mathcal{R}.$$

It follows that

$$\overline{x}\left(k+|\mathcal{R}|\right) \leq \overline{x}\left(k\right) - \left(\prod_{j=0}^{|\mathcal{R}|-1} \eta_i\left(k+j\right)\right) \gamma^{|\mathcal{R}|} \varepsilon_0\left(k\right).$$

Then we have

$$\begin{aligned} V\left(k+|\mathcal{R}|\right) &= \overline{x}\left(k+|\mathcal{R}|\right) - \underline{x}\left(k+|\mathcal{R}|\right) \\ &\leq \overline{x}\left(k\right) - \left(\prod_{j=0}^{|\mathcal{R}|-1} \eta_i\left(k+j\right)\right) \gamma^{|\mathcal{R}|} \varepsilon_0\left(k\right) - \underline{x}\left(k+|\mathcal{R}|\right). \end{aligned}$$

From the safety condition part (5.16), we can easily obtain that $\underline{x}(k+|\mathcal{R}|) \geq$

$\underline{x}(k+|\mathcal{R}|-1) \geq \cdots \geq \underline{x}(k)$. Thus we have

$$\begin{aligned} V\left(k+|\mathcal{R}|\right) &\leq \overline{x}\left(k\right) - \left(\prod_{j=0}^{|\mathcal{R}|-1} \eta_i\left(k+j\right)\right) \gamma^{|\mathcal{R}|} \varepsilon_0\left(k\right) - \underline{x}\left(k\right) \\ &\leq V\left(k\right) - \left(\prod_{j=0}^{|\mathcal{R}|-1} \eta_i\left(k+j\right)\right) \gamma^{|\mathcal{R}|} \frac{1}{2} V\left(k\right) \\ &= \left(1 - \frac{\gamma^{|\mathcal{R}|}}{2}\left(\prod_{j=0}^{|\mathcal{R}|-1} \eta_i\left(k+j\right)\right)\right) V\left(k\right) \\ &\leq \left(1 - \frac{(c\gamma)^{|\mathcal{R}|}}{2}\right) V(k), \end{aligned}$$

where in the last inequality, we used Proposition 5.2.1. A similar result can be obtained in the other case, where the set in (5.21) is empty.

Finally, we can repeat this argument and arrive at

$$V\left(k+l\,|\mathcal{R}|\right) \leq \left(1 - \frac{(c\gamma)^{|\mathcal{R}|}}{2}\right)^{l} V(k).$$

By $0 < \gamma \leq 1$ and $0 < c \leq 1$, we have $V(k) \to 0$ as $k \to \infty$. This completes the proof of the consensus condition part. □

5.3 Resilient consensus problem with asynchronous communication

In this section, we discuss the resilient consensus problem with asynchronous communication. Compared with the problem formulated in Section 5.1, we add an asynchronous communication rule that each normal agent i communicates with its neighbors every N_i steps.

5.3.1 Protocol 2 for asynchronous resilient consensus problem

For Protocol 2, the steps 3) and 4) in MP-MSR algorithm are slightly modified as follows.

Algorithm 5.3.1. *(Protocol 2)*

3. (Local optimization) *Since node i can only receive the last communicated value from its neighbors, so the update rule for the target point is*

$$z_i(k) = x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(\hat{x}_j(k) - x_i(k)\right), \tag{5.22}$$

where $\hat{x}_j(k)$ is the last communicated value.

4. (Local update) *The normal node i not only updates its value, but also determines if it sends its value to the neighbors. So in step 4), node i chooses $u_i(k) = u_i^o(k|k)$ and updates the state $x_i(k+1)$ by* (5.1).

Its communicated value is updated by

$$\hat{x}_i(k+1) = \begin{cases} x_i(k+1) & \text{if } k+1 = lN_i, l \in \mathbb{Z}_+, \\ \hat{x}_i(k) & \text{otherwise.} \end{cases} \tag{5.23}$$

From the update rule, we have

$$x_i(k+1) = x_i(k) + \eta_i'(k) \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(\hat{x}_j(k) - x_i(k)\right), \tag{5.24}$$

where

$$\eta_i'(k) = \frac{u_i^o(k|k)}{\sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(\hat{x}_j(k) - x_i(k)\right)}.$$

Then similarly to the synchronous case, we can show that $\eta_i'(k)$ is lower bounded.

Proposition 5.3.1. *For the MP-MSR algorithm with asynchronous communication, there exists a positive constant* $0 < c < 1$ *such that* $c \leq \eta_i'(k) \leq 1$.

Proof. The proof follows a similar line as that of Proposition 5.2.1. We show the outline as below. The three facts in Lemma 5.2.2 about geometric properties do not change. Note that these three facts hold for a general selecting of $z_i(k)$. It remains to show the following two facts:

1. $|u_i^o(k|k)| \leq \left| \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) \left(\hat{x}_j(k) - x_i(k)\right) \right|$.
2. The sign of $u_i^o(k|k)$ and $\sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) \left(\hat{x}_j(k) - x_i(k)\right)$ are the same.

Since the geometric approach is mainly based on contradiction, in the argument of Proposition 5.2.1, it suffices to replace $x_j(k)$ with $\hat{x}_j(k)$. Then we can obtain the same result. Then, we can establish that $0 < \eta_i'(k) \leq 1$. The next step is to show that there exists a positive constant $0 < c < 1$ such that $\eta_i'(k) \geq c$. The remaining of the proof can be also obtained by replace $x_j(k)$ as $\hat{x}_j(k)$ in Proposition 5.2.1. □

For this problem formulation, we introduce another maximum and minimum of the states of the regular agents: $\overline{\hat{x}}(k) = \max_{i \in \mathcal{R}} \{x_i(k), \hat{x}_i(k)\}$ and $\underline{\hat{x}}(k) = \min_{i \in \mathcal{R}} \{x_i(k), \hat{x}_i(k)\}$. The safety interval $\mathcal{S}$ is chosen as $\mathcal{S} = [\underline{\hat{x}}(0), \overline{\hat{x}}(0)]$. Then we have the following result.

Theorem 5.3.1. *Under the F-total malicious model, the normal agents with MP-MSR under asynchronous communication will reach resilient consensus if and only if the underlying graph is* $(F+1, F+1)$*-robust. The safety interval is given by* $\mathcal{S} = [\underline{\hat{x}}(0), \overline{\hat{x}}(0)]$.

Proof. Since we use the old data $\hat{x}_j(k)$, the proof in Theorem 5.2.1 cannot be directly used here, and thus we conduct a proof with some modifications. The safety interval part is similar as that in Theorem 5.2.1. We replace the role of $\underline{x}(k)$ and $\overline{x}(k)$ with $\underline{\hat{x}}(k)$ and $\overline{\hat{x}}(k)$, by following a similar analysis, we can obtain that $\mathcal{S} = [\underline{\hat{x}}(0), \overline{\hat{x}}(0)]$ is a safety interval and moreover, $\overline{\hat{x}}(k+1) \leq \overline{\hat{x}}(k)$ and $\underline{\hat{x}}(k+1) \geq \underline{\hat{x}}(k)$.

We discuss consensus condition part in the following. Let $\hat{V}(k) = \overline{\hat{x}}(k) - \underline{\hat{x}}(k)$ and introduce the four sets

$$\begin{aligned}
\overline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right) &= \left\{ j \in \mathcal{V} : x_j\left(k'\right) > \overline{\hat{x}}\left(k\right) - \varepsilon_0\left(k'\right) \right\}, \\
\underline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right) &= \left\{ j \in \mathcal{V} : x_j\left(k'\right) < \underline{\hat{x}}\left(k\right) + \varepsilon_0\left(k'\right) \right\}, \\
\overline{\hat{\mathcal{X}}}\left(k, k', \varepsilon_0\left(k'\right)\right) &= \left\{ j \in \mathcal{V} : \hat{x}_j\left(k'\right) > \overline{\hat{x}}\left(k\right) - \varepsilon_0\left(k'\right) \right\}, \\
\underline{\hat{\mathcal{X}}}\left(k, k', \varepsilon_0\left(k'\right)\right) &= \left\{ j \in \mathcal{V} : \hat{x}_j\left(k'\right) < \underline{\hat{x}}\left(k\right) + \varepsilon_0\left(k'\right) \right\},
\end{aligned} \tag{5.25}$$

where $k' \geq k$. We choose $\varepsilon_0\left(k\right) = (\overline{\hat{x}}(k) - \underline{\hat{x}}(k))/2$. Note that $\overline{\hat{\mathcal{X}}}\left(k, k, \varepsilon_0\left(k\right)\right)$ and $\underline{\hat{\mathcal{X}}}\left(k, k, \varepsilon_0\left(k\right)\right)$ are disjoint. The same holds for $\overline{\mathcal{X}}\left(k, k, \varepsilon_0\left(k\right)\right)$ and $\underline{\mathcal{X}}\left(k, k, \varepsilon_0\left(k\right)\right)$.

From (5.24), we have

$$x_i\left(k+1\right) \leq \overline{\hat{x}}\left(k\right) - \eta_i'\left(k\right) \sum_{j \in \mathcal{M}(k) \setminus \overline{\hat{\mathcal{X}}}} a_{ij}\left(k\right) \varepsilon_0\left(k\right),$$

By the $(F+1, F+1)$-robust graph condition, there exists a normal node having neighbours in $\overline{\hat{\mathcal{X}}}\left(k, k, \varepsilon_0\left(k\right)\right)$. Thus, we have

$$x_i\left(k+1\right) \leq \overline{\hat{x}}\left(k\right) - \eta_i'\left(k\right) \gamma \varepsilon_0\left(k\right). \tag{5.26}$$

We choose $\varepsilon_0\left(k+1\right) = \eta_i'\left(k\right) \gamma \varepsilon_0\left(k\right)$. Then after updating, the current state $x_i(k)$ of normal node i is moved out of $\overline{\mathcal{X}}\left(k, k{+}1, \varepsilon_0\left(k+1\right)\right)$. Note that (5.26) also holds

for the nodes outside $\overline{\mathfrak{X}}(k, k, \varepsilon_0(k))$. Thus the current states of regular nodes cannot move in $\overline{\mathfrak{X}}(k, k+1, \varepsilon_0(k+1))$. After N_i steps, for regular node i, it has to update the communicated value. Since its current state $x_i(k+N_i)$ is still outside $\overline{\mathfrak{X}}(k, k+1, \varepsilon_0(k+1))$, at time $k+N_i$, $\hat{x}_i(k+N_i)$ is outside $\overline{\hat{\mathfrak{X}}}(k, k+1, \varepsilon_0(k+1))$. We denote $\overline{N} = \max\{N_i\}$, then we have that after $\overline{N}$ steps, at least one of the regular nodes has to move its current state $x_i(k+\overline{N})$ outside $\overline{\mathfrak{X}}(k, k+1, \varepsilon_0(k+1))$ and is communicated state $\hat{x}_i(k+\overline{N})$ outside $\overline{\hat{\mathfrak{X}}}(k, k+1, \varepsilon_0(k+1))$.

Repeat this argument and set

$$\varepsilon_0(k+l) = \left(\prod_{j=0}^{l-1} \eta_i'(k+j)\right) \gamma^l \varepsilon_0(k), l = 1, 2, \ldots, |\mathcal{R}|.$$

Then after $\overline{N}|\mathcal{R}|$ steps, one of the pairs of sets

$$\overline{\mathfrak{X}}(k, k+|\mathcal{R}|, \varepsilon_0(k+|\mathcal{R}|)) \text{ and } \overline{\hat{\mathfrak{X}}}(k, k+|\mathcal{R}|, \varepsilon_0(k+|\mathcal{R}|))$$

and

$$\underline{\mathfrak{X}}(k, k+|\mathcal{R}|, \varepsilon_0(k+|\mathcal{R}|)) \text{ and } \underline{\hat{\mathfrak{X}}}(k, k+|\mathcal{R}|, \varepsilon_0(k+|\mathcal{R}|)),$$

will be empty.

We suppose the first pair of sets to be empty. Then, we have

$$\overline{\hat{x}}\left(k+\overline{N}|\mathcal{R}|\right) \leq \overline{\hat{x}}(k) - \left(\prod_{j=0}^{|\mathcal{R}|-1} \eta_i'(k+j)\right) \gamma^{|\mathcal{R}|} \varepsilon_0(k).$$

Hence,

$$\begin{aligned}\hat{V}\left(k+\overline{N}\,|\mathcal{R}|\right) &= \overline{\hat{x}}\,(k+|\mathcal{R}|) - \underline{\hat{x}}\,(k+|\mathcal{R}|) \\ &\leq \overline{\hat{x}}\,(k) - \left(\prod_{j=0}^{|\mathcal{R}|-1}\eta_i'\,(k+j)\right)\gamma^{|\mathcal{R}|}\varepsilon_0\,(k) - \underline{\hat{x}}\,(k+|\mathcal{R}|)\,.\end{aligned}$$

From the safety condition part, we can easily know that $\underline{\hat{x}}(k+|\mathcal{R}|) \geq \underline{\hat{x}}(k+|\mathcal{R}|-1) \geq \cdots \geq \underline{\hat{x}}(k)$. Thus we have

$$\begin{aligned}\hat{V}\left(k+\overline{N}\,|\mathcal{R}|\right) &\leq \overline{\hat{x}}\,(k) - \left(\prod_{j=0}^{|\mathcal{R}|-1}\eta_i'\,(k+j)\right)\gamma^{|\mathcal{R}|}\varepsilon_0\,(k) - \underline{\hat{x}}\,(k) \\ &\leq \hat{V}\,(k) - \left(\prod_{j=0}^{|\mathcal{R}|-1}\eta_i'\,(k+j)\right)\gamma^{|\mathcal{R}|}\frac{1}{2}\hat{V}\,(k) \\ &= \left(1-\frac{\gamma^{|\mathcal{R}|}}{2}\left(\prod_{j=0}^{|\mathcal{R}|-1}\eta_i'\,(k+j)\right)\right)\hat{V}\,(k)\,.\end{aligned}$$

Because of $0 < \gamma \leq 1$ and $c \leq \eta_i'\,(k+j) \leq 1$, $j \in \{1, 2, \cdots, |\mathcal{R}|-1\}$. By an analysis similar to Theorem 5.2.1, we have $\hat{V}(k) \to 0$ as $k \to \infty$. This completes the proof of the consensus condition part. □

5.3.2 Protocol 3 for asynchronous resilient consensus problem

We propose another MPC based algorithm in case of asynchronous communication. The main feature of this approach is that the calculation for the MPC part is reduced. In particular, since each node i communicates with neighbors every N_i steps, we solve the optimization problem only at the beginning of the period $[lN_i, (l+1)N_i]$. Thus the steps 3 and 4 in MP-MSR algorithm are modified as

follows:

Algorithm 5.3.2. *(Protocol 3)*

3. (Local optimization) *At every N_i steps, the regular node i solves the optimization problem* $\min_{U_i(lN_i)} J_i\left(x_i\left(lN_i\right), z_i\left(lN_i\right), U_i\left(lN_i\right)\right)$ *with constraints* (5.1) *and* (5.4)*, where*

$$z_i\left(lN_i\right) = x_i\left(lN_i\right) + \sum_{j\in\mathcal{M}_i(lN_i)} a_{ij}\left(lN_i\right)\left(\hat{x}_j\left(lN_i\right) - x_i\left(lN_i\right)\right).$$

4. (Local update) *We write the optimal solution $U_i^o(k)$ as*

$$U_i^o\left(lN_i\right) = \left[u_i^o\left(lN_i|lN_i\right), u_i^o\left(lN_i+1|lN_i\right), \cdots, u_i^o\left(lN_i+N_i-1|lN_i\right)\right]^T,$$

and we choose our control law as: $u_i(k) = u_i^o(k|lN_i), lN_i \le k < (l+1)N_i$. Then update the state by (5.1)*.*

From the update rule, we have

$$x_i\left((l+1)N_i\right) = x_i\left(lN_i\right) + \eta_i''\left(lN_i\right) \sum_{j\in\mathcal{M}_i(lN_i)} a_{ij}\left(lN_i\right)\left(\hat{x}_j\left(lN_i\right) - x_i\left(lN_i\right)\right),$$

where

$$\eta_i''\left(lN_i\right) = \frac{\sum_{j=0}^{N_i-1} u_i^o\left(lN_i+j|lN_i\right)}{\sum_{j\in\mathcal{M}_i(k)} a_{ij}\left(lN_i\right)\left(\hat{x}_j\left(lN_i\right) - x_i\left(lN_i\right)\right)}.$$

Our analysis follows similarly to the previous two cases.

Proposition 5.3.2. *For the modified MP-MSR algorithm, there exists a positive constant $0 < c < 1$ such that $c \le \eta_i''\left(lN_i\right) \le 1$.*

Proof. Note that in this algorithm, the three geometric properties in Lemma 5.2.2 also hold. The proof for the part of $0 < \eta_i''\left(lN_i\right) \le 1$ follows a similar line as that

in Proposition 5.2.1. It remains to show $c \leq \eta_i''(lN_i)$. Rewrite $\eta_i''(lN_i)$ as

$$\begin{aligned}\eta_i''(lN_i) &= \frac{\sum_{j=1}^{N_i-1} u_i^o(lN_i+j|lN_i)}{\sum_{j\in\mathcal{M}_i(k)} a_{ij}(lN_i)(\hat{x}_j(lN_i)-x_i(lN_i))} + \frac{u_i^o(lN_i|lN_i)}{\sum_{j\in\mathcal{M}_i(k)} a_{ij}(lN_i)(\hat{x}_j(lN_i)-x_i(lN_i))} \\ &= \frac{\sum_{j=1}^{N_i-1} u_i^o(lN_i+j|lN_i)}{\sum_{j\in\mathcal{M}_i(k)} a_{ij}(lN_i)(\hat{x}_j(lN_i)-x_i(lN_i))} + \eta_i'(lN_i).\end{aligned} \tag{5.27}$$

Apply the first property of Lemma 5.2.2. We know that

$$\frac{\sum_{j=1}^{N_i-1} u_i^o(lN_i+j|lN_i)}{\sum_{j\in\mathcal{M}_i(k)} a_{ij}(lN_i)(\hat{x}_j(lN_i)-x_i(lN_i))} \geq 0.$$

Thus we have

$$\eta_i''(lN_i) \geq \eta_i'(lN_i) \geq c.$$

□

Now we can proceed to obtain the main result.

Theorem 5.3.2. *Under F-total malicious model, the normal agents with modified MP-MSR with asynchronous communication reach resilient consensus if and only if the underlying graph is $(F+1, F+1)$-robust. The safety interval is given by $\mathcal{S} = [\underline{\hat{x}}(0), \overline{\hat{x}}(0)]$.*

Proof. The proof follows a similar line as that of Theorem 5.3.1. We replace $\eta_i'(k)$ with $\eta_i''(lN_i)$. Note that finally we obtain the consensus condition

$$\lim_{l\to\infty} V(lN_i) = 0.$$

This indicates that at every N_i time instant, the values of regular agents reach resilient consensus. Then we discuss the times between lN_i and $(l+1)N_i$. For $lN_i < k < (l+1)N_i$, we apply the fact that $X_i^o\left(x_i\left(lN_i\right), U_i^o\left(lN_i\right)\right)$ is an N_i-path from $x_i(lN_i)$ to $z_i(lN_i)$. We can also show that

$$\lim_{l\to\infty} \max\{z_i(lN_i)\} - \min\{z_i(lN_i)\} = 0.$$

Thus we say that

$$\lim_{k\to\infty} V(k) = 0.$$

□

5.4 Numerical example

In this section, we illustrate the proposed resilient consensus approaches based on model predictive control via a numerical example.

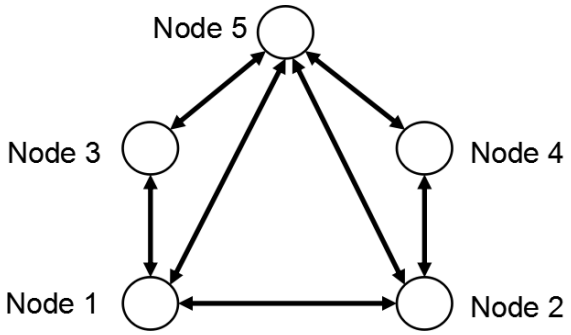


Figure 5.1: Network topology with (2,2)-robustness

Consider the multi-agent system with five nodes whose underlying network is the $(2,2)$-robust graph shown in Fig. 5.1. Node 5 is set to behave in an noncooperative manner, and thus we set $F = 1$. For the model predictive control part, we choose the input constraint as $u_{i,\max} = 0.02$. The parameters of the cost function are set as $\alpha_i = \beta_i = 1$

5.4.1 Simulations in conventional MPC approach

We first check the conventional MPC based consensus algorithm with malicious nodes. We apply two types of malicious behaviors to break the safe condition and consensus condition. In the first simulation, malicious node takes a negative value -1 and tries to mislead the other regular nodes to follow this wrong number. The time responses are depicted in Fig. 5.2, it is clear that all regular nodes are mislead to -1 after 300 steps. In the second simulation, malicious node is set to behave by continuously oscillating their values. The time responses are depicted in Fig. 5.3. It shows that the consensus error between the regular nodes cannot decrease to zero.

5.4.2 Simulations in Protocols 1, 2 and 3

We study three algorithms: (i) MP-MSR algorithm with synchronous communication (Protocol 1). (ii) MP-MSR algorithm with asynchronous communication (Protocol 2). (iii) Modified MP-MSR algorithm with asynchronous communication (Protocol 3).

We first show the performance of protocol 1. The number of receding horizon time steps is set as $N_i = 5$. Then the time responses with state constraints are plotted in Fig. 5.4, where the y-axis represents the value of each agent and the x-axis indicates the common sampling instants k. Though the malicious node 5 (in magenta) exhibits an oscillatory behavior, we observe that the regular nodes reach consensus within the safety interval relatively fast.

The influence of the malicious node to the final consensus result may be present, but is minor. Taking a closer look at the responses, we notice that up to time 100 or so, nodes 2 and 3 (in yellow and red, respectively) fluctuate in their values. This occurs because node 4 (in green) is far from the rest of the regular nodes, and thus, it is considered to be suspicious under the proposed

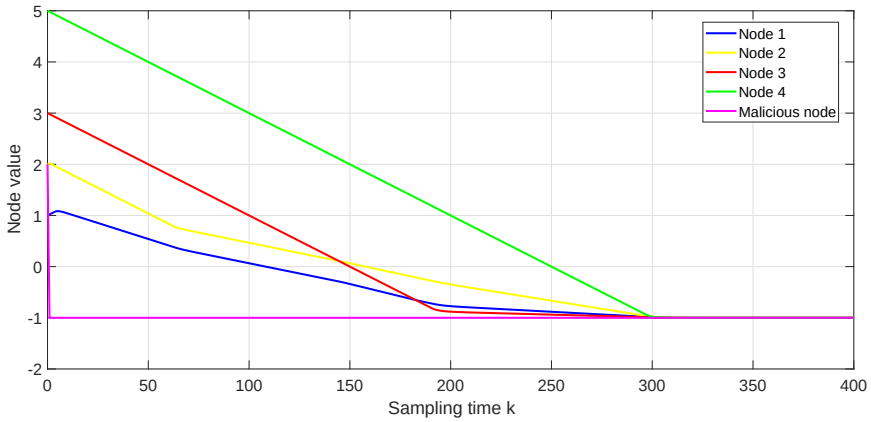


Figure 5.2: Conventional MPC approach with false value malicious node

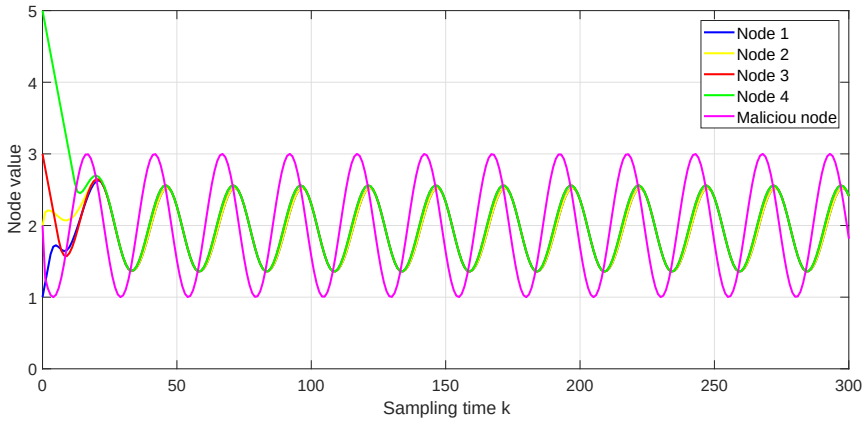


Figure 5.3: Conventional MPC approach with oscillating malicious node

MSR algorithm; instead, the value of the malicious node is used in the updates of the regular nodes.

Further, the control inputs of the regular nodes are shown in Fig. 5.5. It is clear that all of them satisfy the input constraint. This explains the slow convergence of the state of node 4 in Fig. 5.4; its initial state is the furthest from the other regular nodes, and it travels only at the constant velocity determined by $u_{i,\max}$. It is further noticed that the control inputs are optimized since some inputs do not take the maximum value to quickly move to the target point, but rather gently applies the control that decays to zero in a short period of time.

See, for example, the inputs of nodes 2 and 3 around time 100 though these may be partly due to using the value of the malicious node during this time period as discussed above. The calculation time of Protocol 1 is 6.08 seconds for 600 time steps.

In the simulation of Protocol 2, the communication period is different from each node. We choose the communication period as $[5, 6, 7, 8, 9]^T$ and the receding horizon keep the same as communication period, $N_1 = 5, N_2 = 6, N_3 = 7, N_4 = 8, N_5 = 9$. The time responses and control inputs are plotted in Fig. 5.6 and Fig. 5.7. The calculation time of Protocol 2 is 6.17 seconds for 600 time steps.

In the simulation of Protocol 3, we choose the communication period as $[5, 6, 7, 8, 9]^T$ and the receding horizon is kept the same as the communication period, $N_1 = 5, N_2 = 6, N_3 = 7, N_4 = 8, N_5 = 9$. The time responses of the states and the control inputs are plotted in Figs. 5.8 and 5.9. The calculation time of Protocol 3 is 1.07 seconds for 600 time steps.

Comparing the performance for the three protocols, we can see that Protocols 1 and 2 have similar converge speed and calculation time. It is interesting that the control inputs in Protocol 2 are more sensitive to malicious behaviors compared with Protocol 1. The possible reason is that the asynchronous communication may lead to a relatively longtime effect of the malicious agents. Protocol 3 is the least sensitive algorithm for malicious behaviors, and cost the least calculation time. To discuss the factors that infect the dynamics of MP-MSR algorithms may be an interesting problem in the future works.

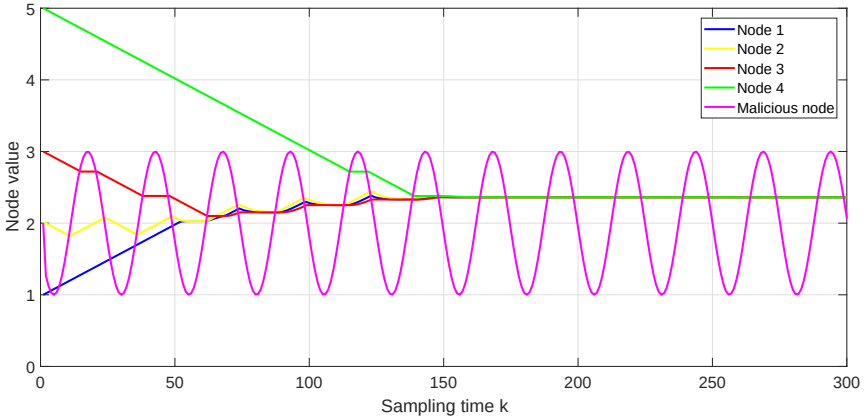


Figure 5.4: Time responses of Protocol 1

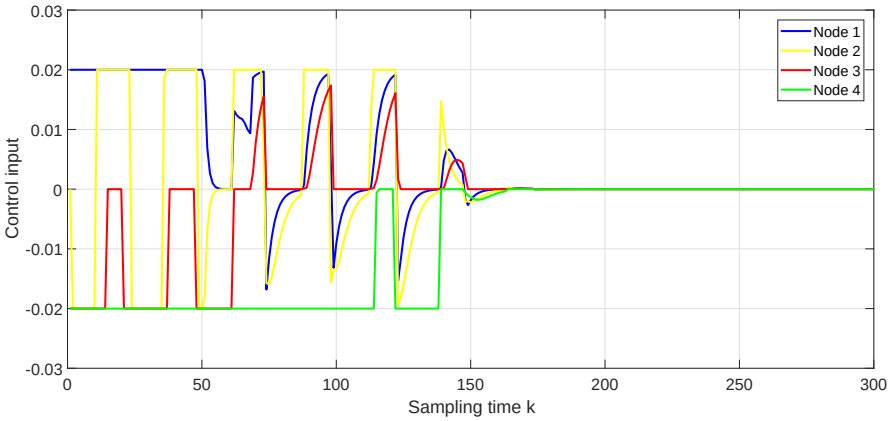


Figure 5.5: Control inputs of Protocol 1

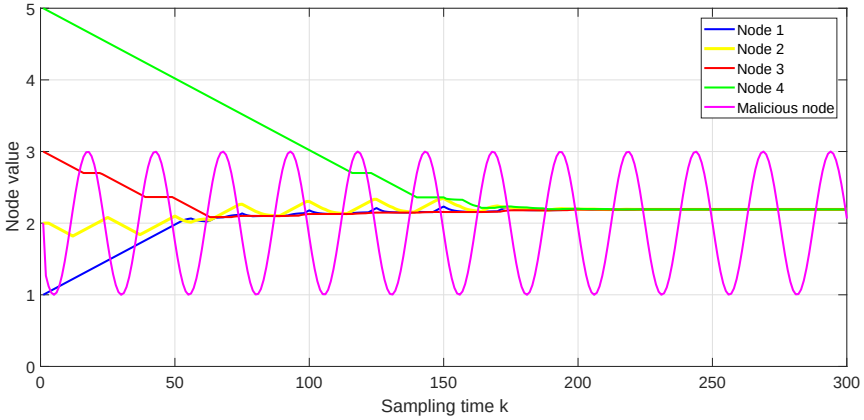


Figure 5.6: Time responses of Protocol 2

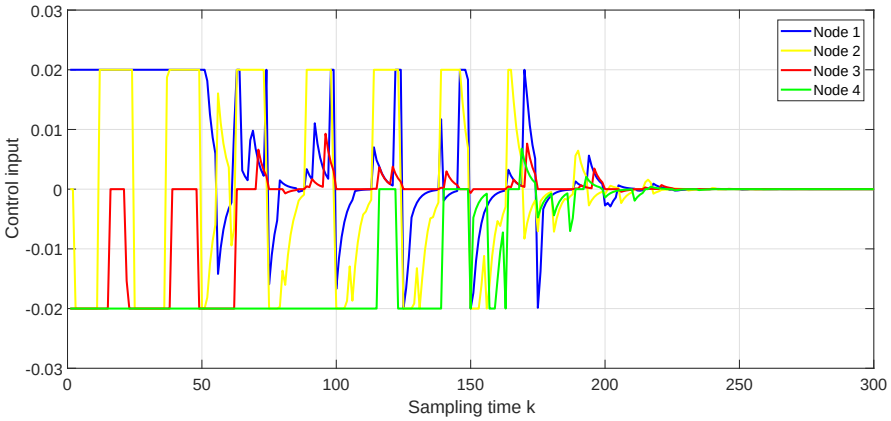


Figure 5.7: Control inputs of Protocol 2

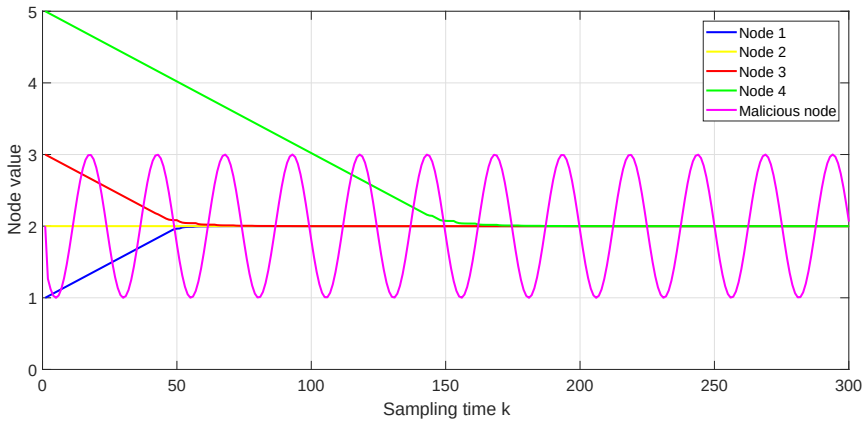


Figure 5.8: Time responses of Protocol 3

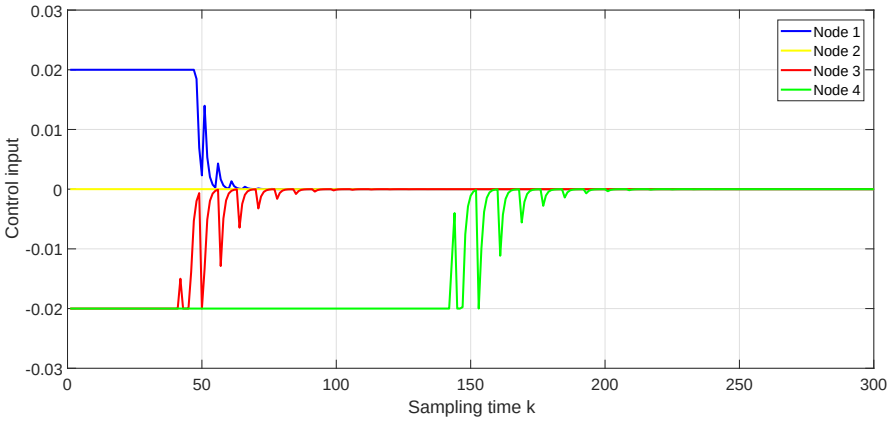


Figure 5.9: Control inputs of Protocol 3

Chapter 6

Resilient Consensus in Mobile Malicious Model

In this chapter, we first discuss the three typical mobile malicious models in the area of computer science ([5; 9; 24]) and apply them to the resilient consensus problem in multi-agent systems. We check that the related results for binary agreement in complete graph can guarantee approximate resilient consensus in multi-agent systems. Moreover, we extend the mobile malicious models to non-complete graphs and propose several novel protocols to solve the resilient consensus problem. In addition, based on Garay's mobile malicious model in [24], we improve the update rules for the cured agents to reduce the necessary connections. Numerical examples are provided to check the efficacy of our results.

6.1 Problem formulation

As seen in previous chapters, the conventional resilient consensus algorithms are mainly based on the following update rule, which is called Mean Subsequence

Reduced (MSR) algorithms in [7; 16; 45]. The typical update rule is written as

$$x_i(k+1) = x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\left(x_j(k) - x_i(k)\right)$$

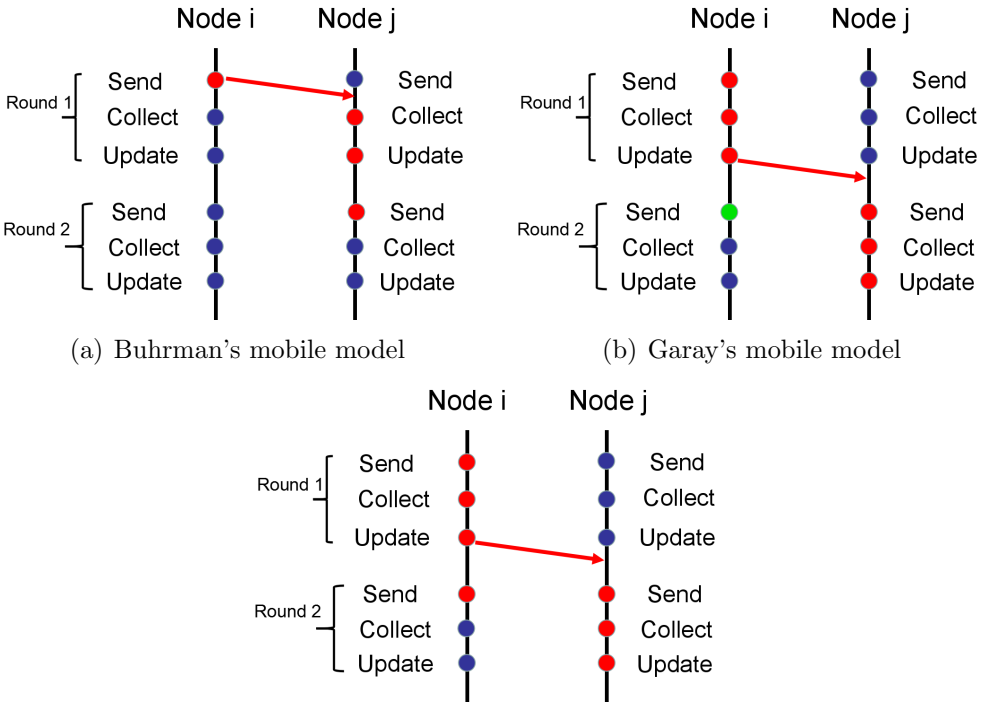
The adversary model is chosen as the F-total malicious, and the related graph structure is $(F+1, F+1)$-robust graph. The analysis is based on the static adversary model and we next show that the mobile adversary can easily destroy resilient consensus.

Consider a graph which is $(2,2)$-robust where one of the agents is attacked and is malicious. From the existing results, we know that if the adversary model is static and the attacked agent remains the same node in the network, the normal agents reach resilient consensus. However in the mobile model, the situation changes. Suppose that, at one time, the adversary moves to a different normal agent, which becomes malicious, and the previous agent recovers and become normal but with a corrupted value left from the attack. At this moment, there are two malicious agents in the network even if the attacker may only manipulate one agent. In such a case, the conventional MSR algorithm cannot guarantee resilient consensus.

Here, we introduce three classes of mobile malicious behaviors. The differences are related to what happens when the adversary moves to another agent and, especially, what the recovering agent is capable to do with the data which may be corrupted. These classes are from the mobile works in [5; 9; 24]. We discuss them one by one. The models are illustrated in Figs. 6.1(a) to 6.1(c).

Note here that the MSR-type algorithms consist of three basic steps in each round [45]: Send, collect, and update. At time k, first, each regular agent i sends its current value $x_i(k)$. Second, it collects the values of the neighbor agents, $x_j(k)$ for $j \in N_i$. Then, after preprocessing to delete some of the neighbor values, the

value is updated to $x_i(k+1)$.



(a) Buhrman's mobile model

(b) Garay's mobile model

(c) Bonnet's mobile model

Figure 6.1: Mobile adversary models

1. Buhrman's model [9]: In this model, the adversary can move from agent i to another agent j only at the sending step (Fig. 6.1(a)). Hence, the cured agent i can immediately collect the neighbors' values and make an update. In this model, the previously infected agents can be cured in one round. As a result, at each round, there are at most F faulty values in the network.

2. Garay's model [24]: The adversary can move from agent i to agent j at any step in a round (Fig. 6.1(b)). In this model, in the first round after becoming cured, agent i is not allowed to send its value to neighbors. That is, agent i is aware that it was infected. Thus, it will only make an update without its corrupted value.

3. Bonnet's model [5]: As in Garay's model, the adversary agent can move at any step during a round (Fig. 6.1(c)). The cured agent i is however not aware of the fact and hence makes the next update as usual. In this case, there are at most $2F$ faulty values in the network: F of them are because of the attacks and the other F are those remaining from the infection in the previous round. Note however that the values of the previously infected agents can become regular in one round.

Our interest in this work is to characterize the necessary networks structure to achieve resilient consensus under these mobile malicious agent models. In the course, we will show that by allowing the cured agents to wait before adopting the regular update rule, we can relax the requirement for the network. This is shown in particular for Garay's model.

4. Our algorithm in Garay's model: The adversary agent can move at any step during round k, but it takes the following two rounds $k+1$ and $k+2$ for the cured agent i to become normal again (Fig. 6.2). In the first round, agent i does not send its value to the neighbors but only makes an update by a rule different from that of regular agents. In the second round, agent i again does not send, but updates its value by an update rule same with regular agents. The values of the previously infected agents become regular in two rounds.

The difference between our algorithm and Garay's original algorithm is that, for the regular agents, they remove fewer values and thus the graph condition can be relaxed. Meanwhile, such regular update rule is more fragile to adversary values. Thus we need the first round, which applies an update rule which removes more values and then we can guarantee that all cured agents are inside a safety area at the end of the first round.

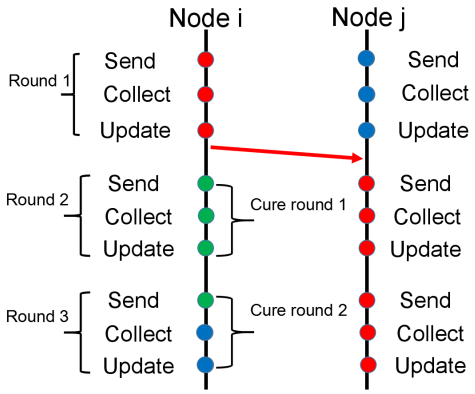


Figure 6.2: Our algorithm in Garay's mobile model

6.2 Protocol 1 for Buhrman's and Garay's models

6.2.1 Modified MSR algorithm 1 (Protocol 1)

Here, we present the first protocol to mitigate the effects of the mobile adversaries, it is a modified version of the MSR algorithm in, e.g., [16; 45]. It will be shown that this protocol is effective to deal with the mobile malicious agents under Buhrman's model and Garay's model.

The Protocol 1 has four steps as shown below. In step 1, the communication part may change depending on the mobile adversary model.

Algorithm 6.2.1. *(Protocol 1) At each round k, each regular agent i executes the following four steps:*

1. (Communication) *Agent i sends its current value $x_i(k)$ to its neighbors according to the mobile adversary model.*

2. (Collecting neighbor information) *Agent i collects the values of neighbors $x_j(k), j \in \mathcal{N}_i$. Then, it sorts the received values and its own value in descending order.*

3. (Deleting suspicious values) *After the sorting, agent i deletes the F largest and F smallest values. The deleted data will not be used in the update of its value. The set of the agents of the remaining values is written as $\mathcal{M}_i(k) \subset \mathcal{N}_i$.*

4. (Local update) *Agent $i \in \mathcal{R}$ updates its value by*

$$x_i(k+1) = a_{ii}(k)x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\, x_j(k), \tag{6.1}$$

where $a_{ii}(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) = 1$ *and* $\gamma \leq a_{ij}(k) \leq 1$

The difference of this algorithm from those in [16; 45] is mainly in step 3). Here, $2F$ values are deleted regardless of the value of agent i, whereas in conventional algorithms, this number depended on the current value of agent i. Specifically, if the value of agent i is among the largest F (resp., the smallest F), then only those greater (resp., smaller) than $x_i(k)$ are deleted. Note that in the current algorithm, agent i might not use its own value.

6.2.2 Convergence of Protocol 1 under Buhrman's model

We establish that with Protocol 1, we can achieve resilient consensus under Buhrman's model. We first present the result for networks in the complete graph forms. Let $\overline{x}(k) = \max\{x_i(k), i \in \mathcal{R}\}$, $\underline{x}(k) = \min\{x_i(k), i \in \mathcal{R}\}$.

Lemma 6.2.1. *Consider an agent network that is represented as a complete graph. Suppose that the model of the mobile adversaries is F-total malicious and follows Buhrman's behaviors. Then, regular agents using Protocol 1 reach resilient consensus if and only if the graph satisfies $|\mathcal{V}| \geq 2F + 1$. The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.*

Proof. The necessity part directly comes from Theorem 6 in [6]. We next show the sufficient part. Based on Theorem 1 (Tables 1 and 2) in [6], with F-total mobile malicious (which is called symmetric in [6]) model, we must show that the MSR algorithm Protocol 1 has the following two properties:

P1 For each regular agent, the recent updated value $x_i(k+1)$ should be inside the range of last regular values $[\underline{x}(k), \overline{x}(k)]$.

P2 The difference between regular agents is strictly smaller than the last time whenever the difference is nonzero, that is, $\overline{x}(k+1) - \underline{x}(k+1) < \overline{x}(k) - \underline{x}(k)$.

Then it follows that the resilient consensus is achieved when $k \to \infty$.

We check the first property P1 by induction. By assumption, $x_j(0) \in [\underline{x}(0), \overline{x}(0)]$ for regular agents. Next, suppose at time k, we have $x_j(k) \in [\underline{x}(k), \overline{x}(k)]$. Suppose that at the beginning of round 1, there are at most F malicious agents. Based on the deleting step and F-total model, we know that if $x_j(k) \notin [\underline{x}(k), \overline{x}(k)]$, then it is deleted by its neighbors during their updates. So based on (6.1), we have that $\forall i \in \mathcal{R}$, $x_i(k+1) \in [\underline{x}(k), \overline{x}(k)]$. For the cured agents in Buhrman's mobile model, the possible corrupted value at round 1 also comes from the previous infected value $x_i(k)$, such values will be deleted because of the deleting step. So for $i \in \mathcal{C}$ we have

$$x_i\left(k+1\right) = a_{ii}(k)\tilde{x}_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}\left(k\right) x_j\left(k\right),$$

where $\tilde{x}_i(k) \in [\underline{x}(k), \overline{x}(k)]$, and we have $x_i\left(k+1\right) \in [\underline{x}(k), \overline{x}(k)]$. The cured agents all become regular agents at the end of round $k+1$.

We next show P2, Suppose that $\underline{x}(k) < \overline{x}(k)$. We observe that for agents that satisfy $x_i(k) = \overline{x}(k)$, there are at most F agents that are larger than $x_i(k)$, which will be deleted in step 3. So we have $\overline{x}(k+1) \leq \overline{x}(k)$. Similarly, $\underline{x}(k+1) \geq \underline{x}(k)$.

Now we show that $\overline{x}(k+1) = \overline{x}(k)$ and $\underline{x}(k+1) = \underline{x}(k)$ are impossible to be met at the same time if $|\mathcal{V}| \geq 2F+1$. Suppose that $\overline{x}(k+1) = \overline{x}(k)$. Then it follows that the set $\underline{\mathcal{X}}(k) = \{j \in \mathcal{V} : x_j(k) < \overline{x}(k)\}$ has to satisfy that $|\underline{\mathcal{X}}(k)| \leq F$. This is because in the complete graph, each regular agent deletes the F small values from the entire network. Because of $|\mathcal{V}| \geq 2F+1$, we know that for the set $\overline{\mathcal{X}}(k) = \{j \in \mathcal{V} : x_j(k) \geq \overline{x}(k)\}$, we have $|\overline{\mathcal{X}}(k)| \geq F+1$. By definition, $\underline{x}(k)$ is the lower bound for regular agents. So for any regular agent i such that $x_i(k) = \underline{x}(k)$, we have that $x_i(k+1) > \underline{x}(k)$ since $|\overline{\mathcal{X}}(k)| \geq F+1$. Thus, we have that $\underline{x}(k+1) > \underline{x}(k)$. As a result, if $|\mathcal{V}| \geq 2F+1$, then $\overline{x}(k+1) < \overline{x}(k)$ or $\underline{x}(k+1) > \underline{x}(k)$ or both have to be satisfied. Thus we have shown P2. □

We note that in the references [40] and [6], the notion for resilient consensus (which is called Byzantine Approximate Agreement in [6]) is slightly different. The consensus condition in these works are given by: For any regular agent $i, j \in \mathcal{R}$, if $k \geq k_f$, then we have $|x_i(k) - x_j(k)| \leq \epsilon$, where k_f is a finite time and ϵ is an arbitrarily small positive real-valued tolerance. In the proof of Lemma 6.2.1, we have shown that our proposed algorithm belongs to the general class of MSR algorithms discussed in [6], which studies the mobile models for the complete graph case. In the following theorem, we extend the result for the non-complete graph case, where a sufficient condition is provided on the graph connectivity for the same algorithm to perform resilient consensus.

Theorem 6.2.1. *Consider an agent network where the model of mobile adversaries is F-total malicious and follows Buhrman's behaviors. Then, regular agents using Protocol 1 reach resilient consensus if the following conditions are satisfied:*

C1. $|\mathcal{V}| \geq 4F+4$.

C2. There exists a number N such that $N \leq |\mathcal{V}|/2$ and for any N-agent subgraph, the agents inside have at least $2F+1$ neighbors from the subgraph.

The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.

Proof. Note that condition C1 is necessary for condition C2 to hold for some N. The safety condition part is obvious from the proof of Lemma 6.2.1. hence, we must prove the consensus condition part. We first introduce two sets:

$$\overline{\mathcal{X}}(k, k', \varepsilon(k')) = \{j \in \mathcal{V} : x_j(k') > \overline{x}(k) - \varepsilon(k')\}, \tag{6.2}$$

$$\underline{\mathcal{X}}(k, k', \varepsilon(k')) = \{j \in \mathcal{V} : x_j(k') < \underline{x}(k) + \varepsilon(k')\}, \tag{6.3}$$

where $k' \geq k$. We choose $\varepsilon(k) = (\overline{x}(k) - \underline{x}(k))/2$. Let $\overline{\mathcal{X}}$ be the shorthand notation for $\overline{\mathcal{X}}(k, k, \varepsilon(k))$, and $\underline{\mathcal{X}}$ for $\underline{\mathcal{X}}(k, k, \varepsilon(k))$. Then there are mainly two cases for discussing the agents in $\overline{\mathcal{X}}$:

1. $|\mathcal{V} \setminus \overline{\mathcal{X}}| \geq N$ and $|\overline{\mathcal{X}}| < N$.
2. $|\mathcal{V} \setminus \overline{\mathcal{X}}| \geq N$ and $|\overline{\mathcal{X}}| \geq N$.

We note that for case $|\mathcal{V} \setminus \overline{\mathcal{X}}| < N$, we have $|\underline{\mathcal{X}}| < N$ and $|\mathcal{V} \setminus \underline{\mathcal{X}}| \geq N$, which is similar to case 1 corresponding to the agents in $\underline{\mathcal{X}}$.

For case 1, we first show that $\forall i \in \overline{\mathcal{X}}$, agent i always has at least $2F + 1$ neighbors from $\mathcal{V} \setminus \overline{\mathcal{X}}$. Take any one agent $i \in \overline{\mathcal{X}}$ and also $N - 1$ agents from $\mathcal{V} \setminus \overline{\mathcal{X}}$. From the condition C2 on subgraphs, we know that agent i receives values from at least $2F+1$ neighbors in $\mathcal{V} \setminus \overline{\mathcal{X}}$. Moreover, the set $\mathcal{M}_i(k) \cap \left(\mathcal{V} \setminus \overline{\mathcal{X}}\right)$ is not empty under Protocol 1 at all rounds k. Then under Protocol 1, the update rule (6.1) can be rewritten as

$$x_i(k+1) = \sum_{j \in \mathcal{M}_i(k) \cap \left(\mathcal{V} \setminus \overline{\mathcal{X}}\right)} a_{ij}(k)\, x_j(k) + \sum_{j \in \mathcal{M}_i(k) \cap \overline{\mathcal{X}}} a_{ij}(k)\, x_j(k) \leq \overline{x}(k) - \gamma\varepsilon(k). \tag{6.4}$$

For the agents $i \in \mathcal{V} \setminus \underline{\mathcal{X}}$, we first introduce the following fact: By the condition C2,

since for any N-agent subgraph, the agents inside have at least $2F+1$ neighbors from the subgraph, this condition also holds for any N'-agent subgraph, where $N \leq N' \leq |\mathcal{V}|$.

Since $|\mathcal{V} \setminus \overline{\mathcal{X}}| \geq N$, by this fact we know that the inequality (6.4) still holds because that $\mathcal{M}_i(k) \cap \left(\mathcal{V} \setminus \overline{\mathcal{X}}\right)$ is not empty under Protocol 1 and thus for all regular agents $i \in \mathcal{R}$, we have

$$x_i\left(k+1\right) \leq \overline{x}(k) - \gamma\varepsilon(k).$$

For case 2, by applying similar analysis, we know that for all regular agents, they have at least $2F+1$ neighbors from $\mathcal{V} \setminus \overline{\mathcal{X}}$, and $\mathcal{M}_i(k) \cap \left(\mathcal{V} \setminus \overline{\mathcal{X}}\right)$ is not empty under Protocol 1. Thus (6.4) also holds for this case. By choosing $\varepsilon(k+1) = \gamma\varepsilon(k)$, we can guarantee that all regular agents are outside the set $\overline{\mathcal{X}}(k, k+1, \varepsilon(k+1))$ at time $k+1$.

Similarly, for the case $|\mathcal{V} \setminus \overline{\mathcal{X}}| < N$, we have that all regular agents are outside the set $\underline{\mathcal{X}}(k, k+1, \varepsilon(k+1))$ at time $k+1$.

So at time $k+1$, at least one of the sets $\overline{\mathcal{X}}(k, k+1, \varepsilon(k+1))$ and $\underline{\mathcal{X}}(k, k+1, \varepsilon(k+1))$ does not contain any regular agents. We suppose the first set that applies to. Then we have

$$\begin{aligned} V\left(k+1\right) &= \overline{x}(k+1) - \underline{x}(k+1) \\ &\leq \overline{x}(k) - \gamma\varepsilon(k) - \underline{x}(k+1) \\ &\leq \overline{x}(k) - \gamma\varepsilon(k) - \underline{x}(k) \\ &\leq V(k) - \gamma\varepsilon(k) \\ &= \left(1 - \frac{\gamma}{2}\right) V(k). \end{aligned} \tag{6.5}$$

The same bound holds for the other case. Thus we have $V(k) \to 0$ when $k \to$

∞. □

We note that for the class of graph discussed in Theorem 6.2.1, there is a common property for them. The following lemma gives a necessary condition for such graph.

Lemma 6.2.2. *Assume that the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ satisfies the condition C1 and C2 of Theorem 6.2.1 for some F. Then the number of neighbors for each agent in the graph satisfies $|\mathcal{N}_i| > |\mathcal{V}|/2$.*

Proof. The proof comes from a contradiction. We suppose that $|\mathcal{N}_i| \leq |\mathcal{V}|/2$ for some $i \in \mathcal{V}$ and the condition C2 is satisfied. From $|\mathcal{N}_i| \leq |\mathcal{V}|/2$ we know that there are more than $|\mathcal{V}|/2$ agents that are not the neighbors of agent i. Since $N \leq |\mathcal{V}|/2$, we can take agent i and another $N-1$ agents from the agents that are not the neighbor of agent i. Then we obtain an N-agent subgraph and it is clear that agent i does not have any neighbor from this subgraph, which is a contradiction with C1. □

Based on Lemma 6.2.2, the following corollary gives a class of graphs that can guarantee resilient consensus under the adversary model in Theorem 6.2.1. Compared with the graph discussed in Theorem 6.2.1, this class of graphs is easier to check.

Corollary 6.2.1. *Consider an agent network where the model of mobile adversaries is F-total malicious and follows Buhrman's behaviors. Then, regular agents using Protocol 1 reach resilient consensus if the following conditions are satisfied:*

C1. $|\mathcal{V}| \geq 4F + 4$.

C2. For every agent i, the number of neighbors $|\mathcal{N}_i| \geq 2F + 1 + |\mathcal{V}|/2$.

The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.

Proof. We would like to show that graphes satisfying C1 and C2 form is a subset of the graphs discussed in Theorem 6.2.1. Delete any $\lceil |\mathcal{V}|/2 \rceil$ agents in the graph, we know that the remaining subgraph is $\lfloor |\mathcal{V}|/2 \rfloor$-agent subgraph. Based on the second condition, we know that every agent inside has at least $2F + 1$ neighbors from the subgraph. Thus we choose $N = \lfloor |\mathcal{V}|/2 \rfloor$, and we have that for any N-agent subgraph, the agents inside have at least $2F + 1$ neighbors from the subgraph. □

Corollary 6.2.1 gives a general sketch for the non-complete graphs under Buhrman's model. In order to guarantee resilient consensus, we have to guarantee that each agent has neighbors more than half of the total agents. Note that there is a gap between the graph conditions discussed in Theorem 6.2.1 and Lemma 6.2.1. The reason is that we obtain a necessary and sufficient condition for complete graphs in Lemma 6.2.1. However, in Theorem 6.2.1, it is a sufficient condition for non-complete graphs. To find a necessary condition for non-complete graph is a challenging problem and we will study it in future works.

6.2.3 Convergence of Protocol 1 under Garay's model

For Garay's mobile model, the only difference with Buhrman's mobile model is that the cured agents cannot send their values in the curing round. This behavior can be considered as F-total jamming behavior, and then at each round, we have F-total malicious agents and F-total jamming agents in the worst case. Based on an analysis similar to that in Lemma 6.2.1, we have the following lemma for networks in the complete graph forms.

Lemma 6.2.3. *Consider an agent network that is represented as a complete graph. Suppose that the model of the mobile adversaries is F-total malicious and follows Garay's behaviors. Then, regular agents using Protocol 1 reach resilient*

consensus if and only if the graph satisfies $|\mathcal{V}| \geq 3F + 1$. *The safety interval is given by* $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.

In Garay's model, there may be F cured agents that are not allowed to send their values to neighbors. For this reason, in Protocol 1, regular agent i deletes $2F$ neighbor agents in step 3. The cured agents may also be inside the remaining neighbor set $\mathcal{M}_i(k)$. So compared with Buhrman's model, which is discussed in Lemma 6.2.1, we need F more neighbors in Garay's model. This argument also holds for the results extended for non-complete graphs later. We thus obtain the following theorem.

Theorem 6.2.2. *Consider an agent network where the model of mobile adversaries is F-total malicious and follows Garay's behaviors. Then, regular agents using Protocol 1 reach resilient consensus if the following conditions are satisfied:*

C1. $|\mathcal{V}| \geq 6F + 4$.

C2. There exists a number N such that $N \leq |\mathcal{V}|/2$ and for any N-agent subgraph, the agents inside have at least $3F + 1$ neighbors from the subgraph.

The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.

As we discussed, the condition C2 in Theorem 6.2.2 requires F more neighbors than that in Theorem 6.2.1. Moreover, condition C1 is necessary for condition C2.

In the following corollary, we give a class of graphs which are easy to check and can also guarantee resilient consensus.

Corollary 6.2.2. *Consider an agent network where the model of mobile adversaries is F-total malicious and follows Garay's behaviors. Then, regular agents using Protocol 1 reach resilient consensus if the following conditions are satisfied:*

C1. $|\mathcal{V}| \geq 6F + 4$.

C2. For every agent i*, the number of neighbors* $|\mathcal{N}_i| \geq 3F + 1 + |\mathcal{V}|/2$.

The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.

The results for Protocol 1 under Garay's model are slightly different from the ones under Buhrman's model. Generally, the graph condition under Garay's model is stricter than Buhrman's model because of the cured agents' behavior. Here we would like to highlight that the adversary's behavior in Garay's model is more powerful since the adversary agent can move at any step. By contrast, in Buhrman's model, the adversary agent can only move at the send step.

We note that in Garay's model, once the adversary agent moves away, the cured agents know immediately that they have been infected and then avoid sending their values to neighbors. However, in some cases in practice, this feature cannot be guaranteed. For example, there is no fault detection algorithm in the system. To deal with this problem, We discuss Bonnet's mobile model, which does not need cured agents detection. Meanwhile, we propose another protocol to solve the resilient consensus problem for Bonnet's mobile model.

6.3 Protocol 2 for Bonnet's model

6.3.1 Modified MSR algorithm 2 (Protocol 2)

Here, we present a resilient protocol for Bonnet's mobile model, which is a more powerful mobile adversary model. In Bonnet's model, since the cured agents do not know that they were infected, they send corrupted values during the curing round. Then the cured agents can be considered as F-total malicious and at each round, the regular agents may receive at most $2F$ corrupted values (since additional F corrupted values come from the adversary agents). So in Bonnet's

model, we slightly modify the deleting step 3. In particular, the values to be deleted are charged from the F largest and F smallest values to the $2F$ largest and $2F$ smallest values. We would like to show that this protocol is effective to deal with the mobile malicious agents under Bonnet's model.

Protocol 2 consists of four steps as follows.

Algorithm 6.3.1. *(Protocol 2) At each round k, each regular agent i executes the following four steps:*

1. (Communication) *Agent i sends its current value $x_i(k)$ according to the mobile adversary model.*

2. (Collecting neighbor information) *Agent i collects the values of neighbors $x_j(k), j \in \mathcal{N}_i$. Then, it sorts the received values and its own value in descending order.*

3. (Deleting suspicious values) *After the sorting, agent i deletes the $2F$ largest and $2F$ smallest values. The deleted data will not be used in the update of its value. The set of remaining values is written as $\mathcal{M}_i(k) \in \mathcal{N}_i$.*

4. (Local update) *Agent $i \in \mathcal{R}$ updates its value by*

$$x_i(k+1) = a_{ii}(k)x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\, x_j(k), \tag{6.6}$$

where $a_{ii}(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) = 1$ and $\gamma \leq a_{ij}(k) \leq 1$

6.3.2 Convergence of Protocol 2 under Bonnet's model

Since we remove more neighbors in Protocol 2, we need more neighbors compared with the related graphs in Protocol 1. By an analysis similar to that in Lemma 6.2.1, Theorem 6.2.1 and Corollary 6.2.1, we have the following results.

Lemma 6.3.1. *Consider an agent network that is represented as a complete graph. Suppose that the model of the mobile adversaries is F-total malicious and follows Bonnet's behaviors. Then, regular agents using Protocol 2 reach resilient consensus if and only if the graph satisfies $|\mathcal{V}| \geq 4F + 1$. The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.*

Theorem 6.3.1. *Consider an agent network where the model of mobile adversaries is F-total malicious and follows Bonnet's behaviors. Then, regular agents using Protocol 2 reach resilient consensus if the following conditions are satisfied:*

C1. $|\mathcal{V}| \geq 8F + 4$.

C2. There exists a number N such that $N \leq |\mathcal{V}|/2$ and for any N-agent subgraph, the agents inside have at least $4F + 1$ neighbors from the subgraph.

The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.

Corollary 6.3.1. *Consider an agent network where the model of mobile adversaries is F-total malicious and follows Bonnet's behaviors. Then, regular agents using Protocol 2 reach resilient consensus if the following conditions are satisfied:*

C1. $|\mathcal{V}| \geq 8F + 4$.

C2. For every agent i, the number of neighbors $|\mathcal{N}_i| \geq 4F + 1 + |\mathcal{V}|/2$.

The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.

By comparing our results for Bonnet's model with Burhman's model, we notice that neither of them requires the functionality to detect the cured agents for reaching resilient consensus. The only difference is that in Burhman's model, the adversary agent can move at the send step, while in Bonnet's model, the adversary agent can move at any step. It is obvious that Bonnet's model is more powerful. Thus we need a more strictive protocol to guarantee resilient consensus.

Based on the results for Bonnet's model, we observe that each agent needs $2F$ more neighbors compared with Burhman's model.

Next, we compare Bonnet's model with Garay's model. It is easy to see that the adversary agent can move at any step in both Bonnet's model and Garay's model. The difference comes from the defender's viewpoint. In Garay's model, if a regular agent is influenced by an adversary agent, it becomes aware that it was infected as soon as the adversary moves away.

However, in contrast, in Bonnect's model, but the regular agents will never be aware of being infected. Since there is no scheme for detection. So we say that the defender is more capable in Garay's model. Compared with Garay's model, we find that the related results in Bonnet's model typically need F more neighbors.

As discussed above, we can find that the graph conditions are related to the adversaries' power and defender' power. We summarize the differences among the three models in Table 6.1.

Table 6.1: Differences among the three models

Models	Adversary	Defender	Complete graph condition	Non-complete graph condition
Burhman's	weak	weak	$\|\mathcal{V}\| > 2F$	$\|\mathcal{V}\| \geq 4F + 4$
Garay's	strong	strong	$\|\mathcal{V}\| > 3F$	$\|\mathcal{V}\| \geq 6F + 4$
Bonnet's	strong	weak	$\|\mathcal{V}\| > 4F$	$\|\mathcal{V}\| \geq 8F + 4$

We note that in Garay's model, the cured agents are guaranteed to become regular in one round. However, to guarantee this feature, we have to design the update rules carefully and such update rules may requires a more conservative graph condition. If we can extend the one curing round in Garay's model to multiple rounds, then it is possible for us to find another class of algorithms to guarantee resilient consensus under more relaxed graphs. The following section follows this idea. We extend the curing to two rounds, and call them cure rounds

1 and 2. There are different update rules in these rounds. The cured agents in both cure rounds do not send their values. For this model, we propose another protocol to achieve resilient consensus.

6.4 Protocol 3 for Garay's model

In this section, we consider Garay's model from a different viewpoint and develop another resilient consensus algorithm, Protocol 3. Over Protocol 1 for the save mobile malicious model, it has an advantage with respect to necessary network connectivity. In Garay's mobile model, the regular agents have the ability to detect whether they are infected or not. Based on this property, we would like to introduce different update rules for regular agents and cured agents to reduce the necessary graph connections. The values of regular agents can be directly used since they are healthy. However, the cured agents in cure round 1, whose values are still infected, have to ignore their own values in the updates to guarantee the security.

Algorithm 6.4.1. *(Protocol 3) At each round k, each regular agent i executes the following six steps:*

1. (Cure round check) *Agent i checks if it is a cured agent or not. If it is cured agent, check the cure round.*

2. (Communication) *Regular agents send the updated value to all neighbors while cured agents do not send.*

3. (Collecting neighbor information) *Regular and cured agents collect the neighbors' value $x_j(k)$, $j \in \mathcal{N}_i$, and sort them from the largest to the smallest (including its own value $x_i(k)$).*

4. (Deleting behavior for agents in $\mathcal{C}_1(k)$) *Agent i removes the F largest and F smallest values. The deleted data is considered as suspicious data and will not be used in the following local updates. The set of the remaining values are written by $\mathcal{M}_i(k) \subset \mathcal{N}_i$.*

5. (Deleting behavior for regular agents and agents in $\mathcal{C}_2(k)$) *Comparing with $x_i(k)$, agent i removes the F largest and F smallest values from its neighbors. If the number of values larger or smaller than $x_i(k)$ is less than F, then all of them are removed. The set of the remaining values are written by $\mathcal{M}_i(k) \subset \mathcal{N}_i$.*

6. (Local update) *Regular and cured agent updates its value by*

$$x_i(k+1) = a_{ii}(k)x_i(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k)\, x_j(k), \tag{6.7}$$

where $a_{ii}(k) + \sum_{j \in \mathcal{M}_i(k)} a_{ij}(k) = 1$, $\gamma \leq a_{ij}(k) \leq 1$. Note that for agents in cure round 1, $a_{ii}(k)$ may be zero.

On the other hand, for the regular agents or the cured agents in cure round 2, they apply the conventional MSR update rule. More specifically, the cured agents in cure round 1 apply the modified MSR update rule as in Protocol 1, where the difference is that the regular agents in Protocol 3 delete less connections. We will see that Protocol 3 needs less connections in non-complete graphs. Then we introduce the following theorem.

Theorem 6.4.1. *Consider an agent network where the model of the mobile adversaries is F-total malicious and follows Garay's mobile behaviors. Then, regular agents using Protocol 3 reach resilient consensus if the graph is $(4F+1, 2F+1)$-robust. The safety interval is given by $\mathcal{S} = [\underline{x}(0), \overline{x}(0)]$.*

Proof. We first show the safety condition part. Since the graph is $(4F+1, 2F+1)$-robust, for both regular agents and cured agents, they have more than $4F+1$ neighbors. In Protocol 3, at most $2F$ values are deleted and $2F$ may be missing, and thus the neighbor set $\mathcal{M}_i(k)$ is not empty. According to Garay's model, at the beginning of round 1, there are at most F adversary agents and F cured agents. For regular agents and cured agents in cure round 2, i.e., those in $\mathcal{C}_2(k)$, based on the step 5, we know that if for neighbor j, $x_j(0) \notin [\underline{x}(0), \overline{x}(0)]$, then it is deleted. So based on (6.7), we have that $\forall i \in \mathcal{R}(k) \cup \mathcal{C}_2(k)$, $x_i(1) \in [\underline{x}(0), \overline{x}(0)]$. For the cured agents in $\mathcal{C}_1(k)$, the possible corrupted value at time 0 also comes from the previous infected value $x_i(0)$. Such values will be deleted because of the step 4). Note that for $i \in \mathcal{C}_1(k)$ we have the update rule (6.7), where $a_{ii}(0) \neq 0$ only when $x_i(0) \in [\underline{x}(0), \overline{x}(0)]$. Thus we have $x_i(1) \in [\underline{x}(0), \overline{x}(0)]$, that is, the cured agents in $\mathcal{C}_1(0)$ will be within the safety interval at the end of round 1. Repeat this argument, we can obtain the safety condition.

Next we give the consensus condition. We first discuss the update of regular agents. Let $V(k) = \overline{x}(k) - \underline{x}(k)$ and introduce the two sets

$$\begin{aligned} \overline{\mathfrak{X}}(k, k', \varepsilon_0(k')) &= \{j \in \mathcal{V} : x_j(k') > \overline{x}(k) - \varepsilon_0(k')\}, \\ \underline{\mathfrak{X}}(k, k', \varepsilon_0(k')) &= \{j \in \mathcal{V} : x_j(k') < \underline{x}(k) + \varepsilon_0(k')\}, \end{aligned} \tag{6.8}$$

where $k' = k, k+1, \ldots$. We set the sequence $\varepsilon_0(k')$ by

$$\varepsilon_0(k) = \frac{(\overline{x}(k) - \underline{x}(k))}{2},$$

and

$$\varepsilon_0(k'+1) = \gamma \varepsilon_0(k'), \text{for } k' \geq k.$$

Note that $0 \leq \varepsilon_0(k') \leq \varepsilon_0(k'+1)$ for $k' \geq k$. Thus, $\overline{\mathfrak{X}}(k, k, \varepsilon_0(k))$ and $\underline{\mathfrak{X}}(k, k, \varepsilon_0(k))$

are disjoint and nonempty with at least one regular node in each. Therefore, by the assumption of $(4F+1, 2F+1)$-robust, we have the following three cases:

1. All agents in $\overline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right)$ have at least $4F+1$ neighbors from outside.
2. All agents in $\underline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right)$ have at least $4F+1$ neighbors from outside.
3. The total number of agents in $\overline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right)$ and $\underline{\mathcal{X}}\left(k, k', \varepsilon_0\left(k'\right)\right)$ that have at least $4F+1$ neighbors outside is no smaller than $2F+1$.

We would like to show that case 3 will eventually change into case 1 or 2 in the future time, so we first discuss case 3. We know that there are at least $F+1$ regular and cured agents in $\overline{\mathcal{X}}(k, k, \varepsilon_0(k))$ and $\underline{\mathcal{X}}(k, k, \varepsilon_0(k))$ that have at least $4F+1$ neighbors from outside. We first show the updates for the agents in $\overline{\mathcal{X}}(k, k, \varepsilon_0(k))$. Note that the agents in $\underline{\mathcal{X}}(k, k, \varepsilon_0(k))$ can be similarly analyzed.

Partition the agents in $\mathcal{M}_i(k)$ into two parts

- $\mathcal{M}_i(k) \cap \overline{\mathcal{X}}(k, k, \varepsilon_0(k))$,
- $\mathcal{M}_i(k) \setminus \overline{\mathcal{X}}(k, k, \varepsilon_0(k))$.

We would like to show that, since $\varepsilon_0\left(k+1\right) = \gamma\varepsilon_0\left(k\right)$, the regular and cured agents are moved outside $\overline{\mathcal{X}}\left(k, k+1, \varepsilon_0\left(k+1\right)\right)$ at time $k+1$.

We first discuss the regular agents and cured agents in $\mathcal{C}_2(k)$, which follow the regular deleting rule in step 5). From update rule (6.7), we have

$$x_i\left(k+1\right) = a_{ii}(k)x_i(k) + \sum_{j \in \mathcal{M}_i(k) \cap \overline{\mathcal{X}}} a_{ij}\left(k\right) x_j\left(k\right) + \sum_{j \in \mathcal{M}_i(k) \setminus \overline{\mathcal{X}}} a_{ij}\left(k\right) x_j\left(k\right). \quad (6.9)$$

Note that such agents have at least $4F+1$ neighbors from outside $\overline{\mathcal{X}}(k, k, \varepsilon_0(k))$. There are at most $2F$ jamming agents and we remove at most F agents in $\mathcal{M}_i\left(k\right) \setminus \overline{\mathcal{X}}(k, k, \varepsilon_0(k))$. So the set $\mathcal{M}_i\left(k\right) \setminus \overline{\mathcal{X}}(k, k, \varepsilon_0(k))$ is guaranteed to be

nonempty. Thus we have

$$\begin{aligned} x_i\left(k+1\right) &\leq a_{ii}(k)\overline{x}(k)+\sum_{j\in\mathcal{M}_i(k)\cap\overline{\mathfrak{X}}}a_{ij}\left(k\right)\overline{x}\left(k\right)+\sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathfrak{X}}}a_{ij}\left(k\right)\left(\overline{x}\left(k\right)-\varepsilon_0(k)\right) \\ &\leq \overline{x}(k)-\gamma\varepsilon_0(k). \end{aligned} \tag{6.10}$$

On the other hand for the regular agents or the cured agents in cure round 2 outside $\overline{\mathfrak{X}}(k,k,\varepsilon_0(k))$, since $x_i(k) \leq \overline{x}(k)-\varepsilon_0(k)$ and $a_{ii}(k) \geq \gamma$, we can also guarantee (6.10) from (6.9).

Similar results also hold for $i \in \underline{\mathfrak{X}}(k,k,\varepsilon_0(k))$. So we can see that at least $F+1$ regular and cured agents will be outside of $\overline{\mathfrak{X}}\left(k,k{+}1,\varepsilon_0\left(k+1\right)\right)$ and $\underline{\mathfrak{X}}\left(k,k{+}1,\varepsilon_0\left(k+1\right)\right)$ at time $k+1$.

Next we discuss the cured agents $i \in \overline{\mathfrak{X}}(k,k,\varepsilon_0(k)) \cap \mathcal{C}_1(k)$. These agents apply the deleting rule in step 4). We have two possible update rules

$$x_i\left(k+1\right) = a_{ii}(k)x_i(k)+\sum_{j\in\mathcal{M}_i(k)\cap\overline{\mathfrak{X}}}a_{ij}\left(k\right)x_j\left(k\right)+\sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathfrak{X}}}a_{ij}\left(k\right)x_j\left(k\right). \tag{6.11}$$

and

$$x_i\left(k+1\right) = \sum_{j\in\mathcal{M}_i(k)\cap\overline{\mathfrak{X}}}a_{ij}\left(k\right)x_j\left(k\right)+\sum_{j\in\mathcal{M}_i(k)\setminus\overline{\mathfrak{X}}}a_{ij}\left(k\right)x_j\left(k\right). \tag{6.12}$$

The update rule depends on whether agent i deletes its own value or not. Suppose cured agent i has at least $4F+1$ neighbors outside. There are at most $2F$ values missing since cured agents do not send values and we remove at most $2F$ agents in $\mathcal{M}_i\left(k\right)\setminus\overline{\mathfrak{X}}(k,k,\varepsilon_0(k))$. So the set $\mathcal{M}_i\left(k\right)\setminus\overline{\mathfrak{X}}(k,k,\varepsilon_0(k))$ is guaranteed to be non-empty. Then we have that in both (6.11) and (6.12), it holds

$$x_i\left(k+1\right) \leq \overline{x}(k)-\gamma\varepsilon_0(k). \tag{6.13}$$

Thus, since $\varepsilon_0(k+1) = \gamma\varepsilon_0(k)$, we have that at least $F+1$ regular or cured agents at round k move outside $\overline{\mathcal{X}}(k, k{+}1, \varepsilon_0(k+1))$.

We then discuss the behavior of cured agents in $\mathcal{C}_1(k)$ outside $\overline{\mathcal{X}}(k,k,\varepsilon_0(k))$. For $i \in \mathcal{C}_1(k) \setminus \overline{\mathcal{X}}(k,k,\varepsilon_0(k))$, under Protocol 3, we have two possible update rules (6.11) and (6.12). For update rule (6.11), we have similar argument and then we have (6.10) for such class of cured agents. For update rule (6.12), we note that for such cured agent i, its updated value $x_i(k+1)$ may move inside $\overline{\mathcal{X}}(k, k{+}1, \varepsilon_0(k+1))$, for example, if the set $\mathcal{M}_i(k) \setminus \overline{\mathcal{X}}(k,k,\varepsilon_0(k))$ is empty.

Similar results also hold for $i \in \mathcal{C}_1(k) \setminus \underline{\mathcal{X}}(k,k,\varepsilon_0(k))$. Since there are at most F agents that follow the deleting rule in step 4), we know that at most F cured agents can move inside $\overline{\mathcal{X}}(k, k{+}1, \varepsilon_0(k+1))$ or $\underline{\mathcal{X}}(k, k{+}1, \varepsilon_0(k+1))$.

Summarize the argument so far, we know that at the beginning of round $k+1$, there are at most F cured agents move inside $\overline{\mathcal{X}}(k, k{+}1, \varepsilon_0(k+1))$ or $\underline{\mathcal{X}}(k, k{+}1, \varepsilon_0(k+1))$, and at least $F+1$ regular agents move outside. Hence, the number of such agents in $\underline{\mathcal{X}}(k, k+1, \varepsilon_0(k))$ is smaller than that in $\underline{\mathcal{X}}(k,k,\varepsilon_0(k))$ at least by one agent. Repeating this process, we eventually have that there is some time $k+k_f$ such that less than $F+1$ regular agents inside each of $\overline{\mathcal{X}}(k, k{+}k_f, \varepsilon_0(k+k_f))$ and $\underline{\mathcal{X}}(k, k{+}k_f, \varepsilon_0(k+k_f))$, where k_f is a finite number.

We note that at time $k+k_f$, we can check that $\overline{\mathcal{X}}(k, k{+}k_f, \varepsilon_0(k+k_f))$ and $\underline{\mathcal{X}}(k, k{+}k_f, \varepsilon_0(k+k_f))$ are disjoint and nonempty. By the graph robustness, we have the following three cases:

1. All agents in $\overline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ have at least $4F+1$ neighbors from outside.

2. All agents in $\underline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ have at least $4F+1$ neighbors from outside.

3. The total number of agents in $\overline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ and

$\underline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ that have at least $4F+1$ neighbors outside is no smaller than $2F+1$.

From the above analysis we know that there are less than $F+1$ regular and cured agents inside $\overline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ or $\underline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ at time $k+k_f$. So case 3) cannot be satisfied any more. Case 1) and/or case 2) has to be satisfied. We summarize the updates in case 3) in Fig. 6.3.

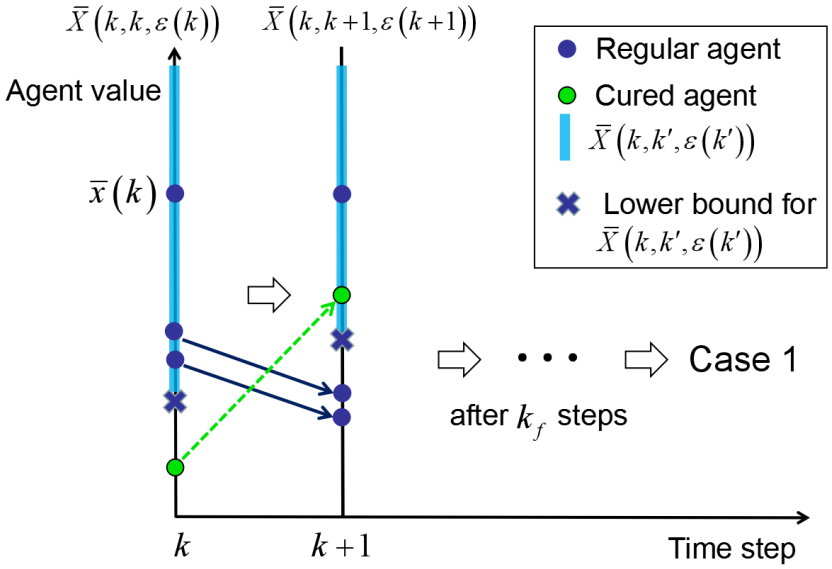


Figure 6.3: Updates for regular and cured agents in case 3

We suppose that case 1) is satisfied. Then we would like to show that the set $\overline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ will be empty for both regular agents and cured agents at the end of round $k+k_f+2$. After the update step in round $k+k_f+1$, we know that all regular and cured agents inside $\overline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ are moved outside. And at most F cured agents in cure round 1 move inside, but such cured agents do not send their values to neighbors at the send step in round $k+k_f+2$. Thus at the send step in round $k+k_f+2$, there is no value sent from the set $\overline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ and then the agents outside cannot move inside anymore. At the beginning of round $k+k_f+2$, the cured agents that moved

inside $\overline{\mathcal{X}}\left(k, k+k_f, \varepsilon_0\left(k+k_f\right)\right)$ in the previous time step are now in cure round 2, and thus they move outside after the update step in this round. Thus we know that at the end of $k+k_f+2$, all regular agents and cured agents are outside $\overline{\mathcal{X}}\left(k, k+k_f, \varepsilon_0\left(k+k_f\right)\right)$. We summarize the updates in case 1) in Fig. 6.4.

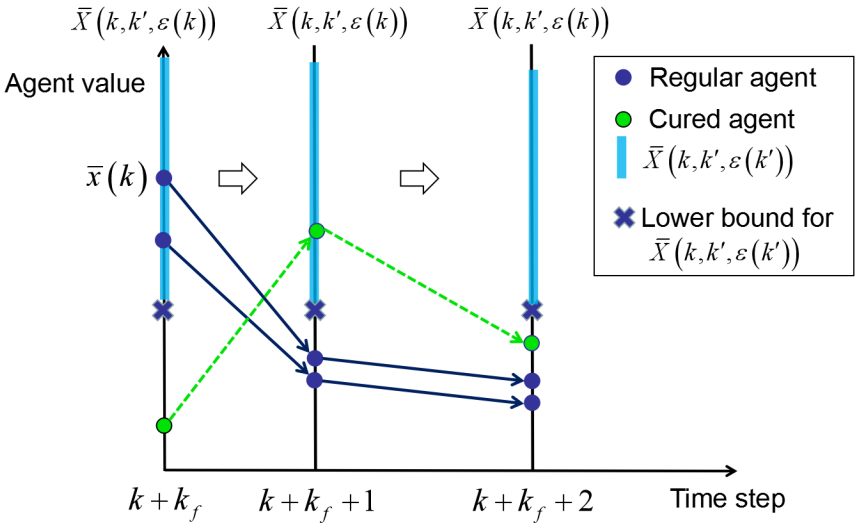


Figure 6.4: Updates for regular and cured agents in case 1

Similar arguments also hold for the set $\underline{\mathcal{X}}\left(k, k+k_f, \varepsilon_0\left(k+k_f\right)\right)$. Then at the end of round $k+k_f+2$, one of the two sets is empty. Here, we suppose $\overline{\mathcal{X}}\left(k, k+k_f, \varepsilon_0\left(k+k_f\right)\right)$ is empty. Then we have

$$x_i\left(k+k_f+2\right) \leq \overline{x}\left(k\right) - \gamma^{k_f}\varepsilon_0\left(k\right), \forall i \in \mathcal{R}.$$

It follows that

$$\overline{x}\left(k+k_f+2\right) \leq \overline{x}\left(k\right) - \gamma^{k_f}\varepsilon_0\left(k\right).$$

We note that $\overline{x}(k)$ is non-increasing and $\underline{x}(k)$ is non-decreasing based on the

update rule (6.7). Then we have

$$\begin{aligned} V\left(k+k_f+2\right) &= \overline{x}\left(k+k_f+2\right) - \underline{x}\left(k+k_f+2\right) \\ &\leq \overline{x}\left(k\right) - \gamma^{k_f}\varepsilon_0\left(k\right) - \underline{x}\left(k\right) \\ &= \left(1-\frac{\gamma^{k_f}}{2}\right)V(k). \end{aligned}$$

Repeating this argument, we have

$$V\left(k+l\left(k_f+2\right)\right) \leq \left(1-\frac{\gamma^{k_f}}{2}\right)^l V(k).$$

Therefore, we have $V(k) \to 0$ as $k \to \infty$. □

The reason that Protocol 3 may guarantee resilient consensus in a relaxed graph mainly comes from the relaxed deleting rules applied to the regular agents. Since we know that the current values of the regular agents are reliable, we can evaluate the other neighbors' value by a specific comparison with the current local value, which is in detail explained in step 5). Compared with Protocol 1, the safe values are more efficiently used in Protocol 3.

Compared with conventional MSR algorithms, there are three main differences in the proof techniques:

- Not only the adversary agents in the graph send corrupted values, but also the cured agents do not send values to neighbors, which is not a regular behavior. Furthermore, the cured agents in $\mathcal{C}_1(k)$ follow an update rule different from regular agents. We have to analyze the convergence of such cured agents separately.

- The behavior of the cured agents in $\mathcal{C}_1(k)$ does not obey the desired update rules for regular agents. Thus the cured agents outside $\underline{\mathfrak{X}}\left(k,k,\varepsilon_0\left(k\right)\right)$ may

move inside the set $\overline{\mathcal{X}}(k, k+1, \varepsilon_0(k+1))$, which does not appear in conventional MSR algorithms analysis. This property leads us to some technical difficulties in the analysis.

- The set $\underline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ is guaranteed to be empty at time $k+k_f+2$, which is two rounds later than the conventional MSR analysis. The reason is the behavior of cured agents. From the proof we know that in the worst case the cured agents will first move inside $\underline{\mathcal{X}}(k, k+k_f, \varepsilon_0(k+k_f))$ and then move outside. In this work, we have to guarantee that the set is empty for both regular and cured agents, while the conventional works only requires the set to be empty for regular agents.

Compared with the non-complete results in Garay's model, the graph condition discussed here is only determined by the total number of adversary agents F. Moreover, we avoid the condition that every regular agent has at least more than $|\mathcal{V}|/2$ edges, which appears in Theorem 6.2.2 and Corollary 6.2.2. Here, we must emphasize that for a given F, if $|\mathcal{V}|$ is large enough, the connectivity in Theorem 6.4.1 may become less than that in Corollary 6.2.2. We give a non-complete graph example that is $(5,3)$-robust, but does not satisfy the conditions in Theorem 6.2.2. This shows that the graph conditions in Theorem 6.4.1 are more relaxed. Suppose that $F = 1$ and $|\mathcal{V}| = 10$. Then we can easily check that only the 10 agent complete graph satisfies the two conditions in Theorem 6.2.2. We can also check that the graph in Fig. 6.5, that each agent has at least 8 neighbors is $(5,3)$-robust graph. As shown in Fig. 6.5, the agents in the dash line circle can fully communicate with each other. The arrows indicate the neighbor information between dash line circles.

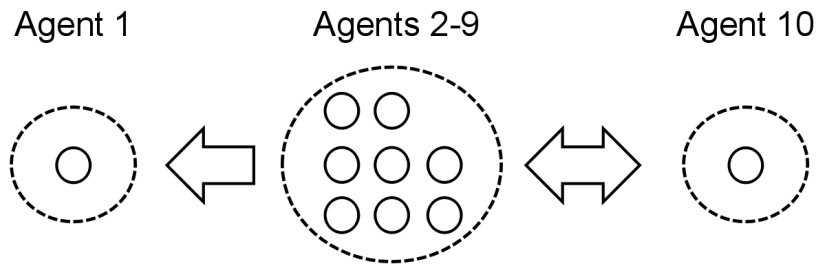


Figure 6.5: An example of $(5, 3)$-robust graph

6.5 Numerical Examples

In this section, we would like to check the performance of conventional MSR algorithm in mobile adversary model. Then, we illustrate the proposed protocols in numerical examples.

The first graph that we consider here is a 14 agent graph, where agent 1 does not have an in-neighbor from agent 14. Agents 2 to 14 do not have in-neighbor from agent 1. The other connections are all connected. We can check that all agents of the underlying graph in Fig. 6.6 have 12 neighbors. For $F = 1$, this graph satisfies the conditions in Theorems 6.2.1, 6.2.2, 6.3.1 and 6.4.1. We would like to check the effectiveness of Protocols 1 and 2 under this graph.

The second graph with 14 agent is shown in Fig. 6.7. We can check that this graph is (5,3)-robust and the agents have at most 9 neighbors (agents 6-14 only have 8 neighbors). Note that this graph does not satisfy the graph conditions in Theorem 6.2.2 under Garay's model. We would like to check the effectiveness of Protocol 3 for Garay's model under this graph.

The initial states are random numbers in the interval of $[0, 8]$, and thus non-negative. The adversary agent always takes a negative number -5 and moves from agent 1 to agent 14.

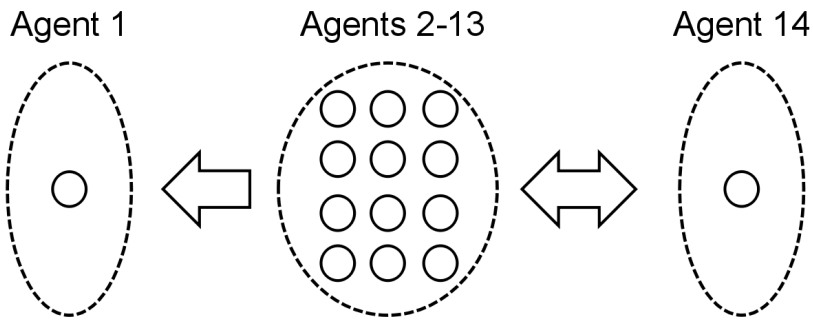


Figure 6.6: Graph 1

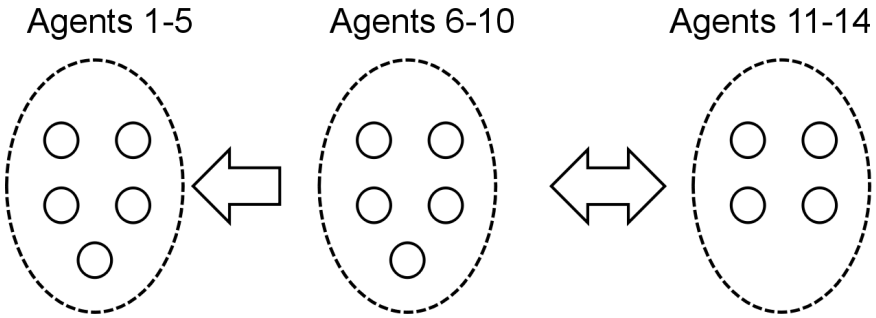


Figure 6.7: Graph 2

6.5.1 Simulations for conventional MSR

We first check the conventional MSR algorithm in mobile adversary models. We apply the Buhrman's mobile model under graph 1 and show the results in Fig. 6.8. It is easy to see that the adversary agent can lead all regular agents to a negative value, which is determined by the adversary agent. We note that the other mobile adversary models can also lead to similar results. So the conventional MSR algorithm cannot guarantee resilient consensus in any of the mobile adversary models.

6.5.2 Simulations for Protocols 1 and 2 in graph 1

We check the effectiveness of the proposed Protocols 1 and 2 in graph 1. We first give the result for Protocol 1 under Burhman's model. The time responses

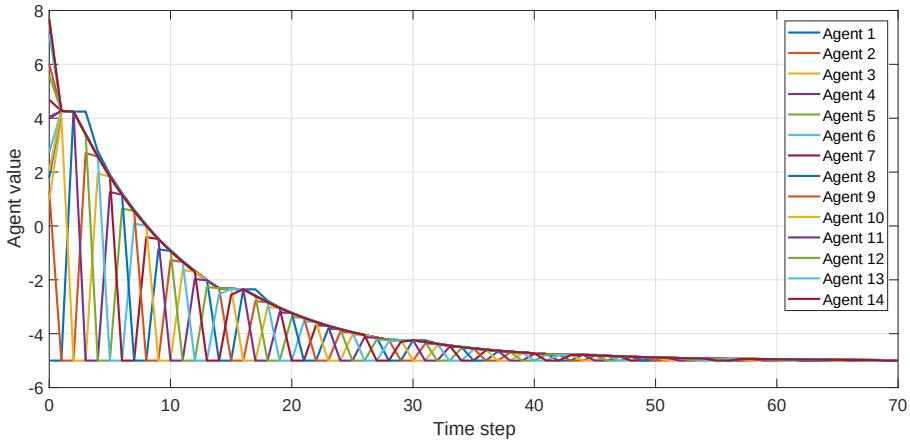


Figure 6.8: Conventional MSR algorithm in Burhman's model

are depicted in Fig. 6.9. From this plot we know that once the adversary agent moves away, the cured agent can be recovered to regular agents in one round. For Protocol 1 in Garay's model, we also have a similar plot. Here, the cured agents do not send their values. The difference is that adversary agent can move at any step in a round and thus we may have at most $2F$ adversary values in one round. For Protocol 2 under Bonnet's model, by removing more neighbor values, we can avoid the cured agent detection in Garay's model, and the time responses are depicted in Fig. 6.10. We found that in these numerical examples, the resilient consensus are reached. Since the underlying graph satisfies the graph conditions in Theorems 6.2.1, 6.2.2, 6.3.1, we have shown the effectiveness of these results.

6.5.3 Simulations for Protocol 3 in graph 2

Next we show that Protocol 3 for Garay's model can solve the resilient consensus problem in a relaxed graph. This simulation is conducted in graph 2, which is (5,3)-robust. We give the results in Fig. 6.11. It is easy to see that Protocol 3 successfully solves the resilient consensus problem. Compared with the results in graph 1, the convergence speed is slower because of the reduced connectivity.

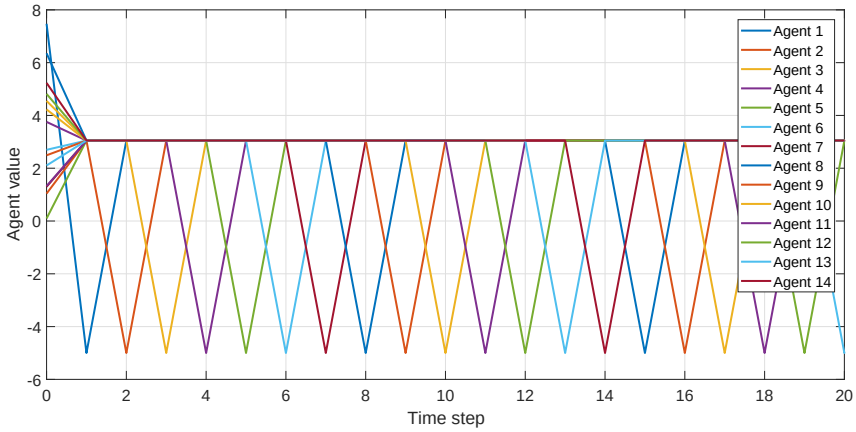


Figure 6.9: Protocol 1 in Burhman's model

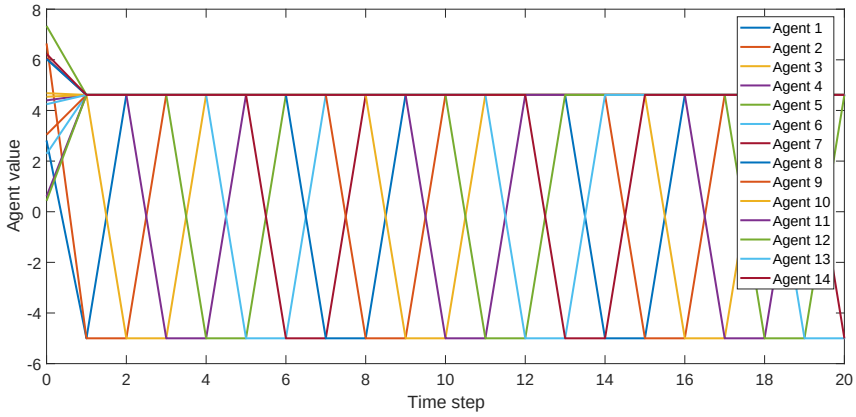


Figure 6.10: Protocol 2 in Bonnet's model

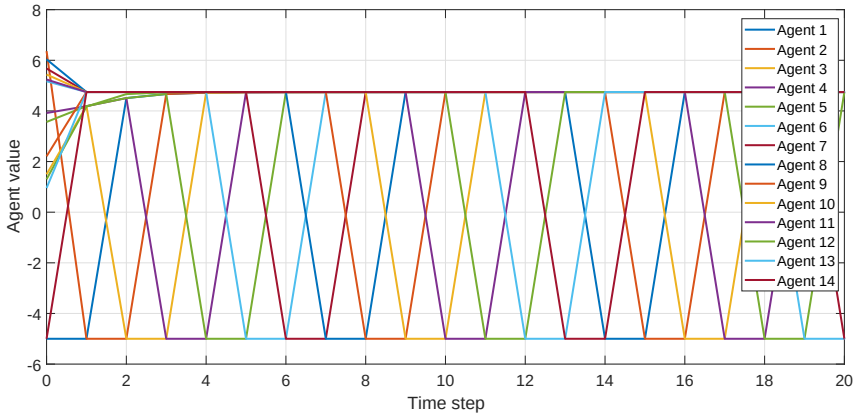


Figure 6.11: Protocol 3 in Garay's model

Chapter 7

Conclusion

7.1 Summary of achievements

The main contribution of this thesis lies in the following four parts:

1. **Event-based resilient consensus protocols with limited communications**

   In Chapter 3, we considered a resilient approach for the multi-agent consensus problem to mitigate the influence of misbehaving agents due to faults and cyber-attacks. Two protocols for the updates of the regular nodes have been proposed, and their convergence properties as well as necessary network structures have been characterized. In both cases, resilient consensus can be achieved with reduced frequencies in communication among agents through event triggering. This is possible at the expense of certain errors in consensus determined by the parameters in the triggering function.

2. **Quantized resilient consensus protocols with limited communications**

   In Chapter 4, we considered the quantized resilient approaches for the multi-agent consensus problem to mitigate the influence of misbehaving agents due to faults and cyber attacks. Two novel protocols for the updates of the

regular nodes have been proposed, and the convergence properties as well as necessary network structures have been characterized. The quantized resilient consensus can be achieved with reduced frequencies in communication among agents through event triggering.

3. **Resilient consensus protocols with input constraint**

   In Chapter 5, we have considered a model predictive based approach for the multi-agent consensus problem in the presence of malicious agents. A model predictive control based resilient protocol has been proposed. It follows the MSR algorithm to mitigate the influence of the misbehaving adversaries, while it optimizes the control by taking account of input constraints. We have established that resilient consensus is reachable if the agent network possesses robust properties as a graph.

4. **Resilient consensus protocols under mobile adversary models**

   In Chapter 6, we have considered resilient protocols for the multi-agent approximate consensus problem to mitigate the influence of mobile misbehaving agents due to faults and cyberattacks. Two protocols are proposed to solve the resilient consensus problem in three conventional mobile adversary models. In addition, we have also proposed a new mobile adversary model and the related resilient consensus protocol. In all cases, resilient consensus can be achieved in certain class of graphs.

7.2 Future directions

Even though resilient control problems have recently become a hot topic and the research has lead to fruitful achievements, there is still much room for improvement of this topic. Some of them such as the two open problems listed in Table 6.1 may be straightforward to solve by exploiting existing results. However,

there are many research directions that require further efforts and need formal formulation and theoretical analysis.

1. **Effect of noise**

   Resilient consensus algorithms with system noises for the event-triggered case is one of the possible directions. Based on different noise models such as Gaussian white noise or unknown but bounded noise, the performance of MSR algorithms can be analyzed by making use of techniques in stochastic or set theories.

2. **More detailed mobile adversary models**

   The existing mobile models focus on the mobile time instant in a round. There are many other aspects such as the possible mobile area, mobile frequency and so on. If the mobile adversary can be formulated in more details, the mobile models can be more realistic and useful. Moreover, epidemic models in computer networks or social networks may provide inspirations for such directions.

3. **Leader-follow consensus in mobile adversary models**

   In Chapter 6, we discussed the approximate resilient consensus problems under mobile adversary models. Another direction is the study the leader-follower resilient consensus problem. If the mobile adversary model has some protected area so that the leaders cannot be infected, then it is desirable to reduce the necessary connections compared with the approximate resilient consensus in Chapter 6.

4. **Resilient control in cooperation with fault detection approaches**

   Even though the major advantage of resilient control is not to use the fault detections, it is meaningful to find a tradeoff between resilient control

approaches and fault detection approaches. In resilient control works, more connections are usually required to guarantee convergence. If a distributed fault detection algorithm is equipped, it should be possible to reduce the connections compared with the conventional resilient control algorithms.

5. **Resilient consensus problem in other backgrounds**

   Many existing resilient consensus works are motivated by applications in wireless sensor networks and mobile robot networks. It is interesting to extend such approaches to other applications such as boolean networks. The malicious behavior of applications may be different, and such difficulties may motivate us to find novel models or approaches to solve them.

6. **Other quantization approaches in quantized resilient consensus** In Chapter 4, we have employed a probabilistic quantizer to solve the quantized resilient consensus problem. Based on the requirements, to study the performance of other quantization approaches such as logarithmic quantizers is also an interesting problem.

References

[1] A. Alam, A. Gattami, and K. H. Johansson. Guaranteeing safety for heavy duty vehicle platooning: Safe set computations and experimental evaluations. *Control Engineering Practice*, 24:33–41, 2014.

[2] C. Alippi. *Intelligence for Embedded Systems*. Springer Verlag, 2014.

[3] T. C. Aysal, M. J. Coates, and M. G. Rabbat. Distributed average consensus with dithered quantization. *IEEE Transactions on Signal Processing*, 56(10):4905–4918, 2008.

[4] A. Beloglazov, J. Abawajy, and R. Buyya. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Generation Computer Systems*, 28(5):755–768, 2012.

[5] F. Bonnet, X. Défago, T. D. Nguyen, and M. Potop-Butucaru. Tight bound on mobile byzantine agreement. *Theoretical Computer Science*, 609:361–373, 2016.

[6] S. Bonomi, A. D. Pozzo, M. Potop-Butucaru, and S. Tixeuil. Approximate agreement under mobile Byzantine faults. *Theoretical Computer Science*, 758:17–29, 2019.

[7] Z. Bouzid, M.G. Potop-Butucarum, and S. Tixeuil. Optimal Byzantine-

resilient convergence in uni-dimensional robot networks. *Teoretical Computer Science*, 411:3154–3168, 2010.

[8] F. Brauer and C. Castillo-Chavez. *Mathematical Models in Population Biology and Epidemiology*. Springer, 2001.

[9] S. Buhrman, J. A. Garay, and J. H. Hoepman. Optimal resiliency against mobile faults. In *Proceedings of the 25th International Symposium on Fault-Tolerant Computing*, pages 83–88, 1995.

[10] K. Cai and H. Ishii. Quantized consensus and averaging on gossip digraphs. *IEEE Transactions on Automatic Control*, 56(9):2087–2100, 2011.

[11] R. Carli, F. Fagnani, P. Frasca, and S. Zampieri. Gossip consensus algorithms via quantized communication. *Automatica*, 46(1):70–80, 2010.

[12] A. Cetinkaya, H. Ishii, and T. Hayakawa. Networked control under random and malicious packet losses. *IEEE Transactions on Automatic Control*, 62(5):2434–2449, 2017.

[13] A. W. Colombo, T. Bangemann, S. Karnouskos, J. Delsing, P. Stluka, R. Harrison, F. Jammes, and J. Lastra. *Industrial Cloud-based Cyber-Physical Systems: The IMC-AESOP Approach*. Springer Verlag, 2014.

[14] X. Défago, M. Potop-Butucaru, and S. Tixeuil. Fault-tolerant mobile robots. *Distributed Computing by Mobile Entities*, 11340:234–251, 2019.

[15] X. Défago, M. G. Potop-Butucaru, and P. Raipin-Parvédy. Self-stabilizing gathering of mobile robots under crash and byzantine failures. *DOI: 10.1007/s00446-019-00359-x, Distributed Computing, Springer*, 2019, to appear.

[16] S. M. Dibaji and H. Ishii. Consensus of second-order multi-agent systems in the presence of locally bounded faults. *Systems & Control Letters*, 79:23–29, 2015.

[17] S. M. Dibaji and H. Ishii. Resilient consensus of second-order agent networks: Asynchronous update rules with delays. *Automatica*, 81:123–132, 2017.

[18] S. M. Dibaji, H. Ishii, and R. Tempo. Resilient randomized quantized consensus. *IEEE Transactions on Automatic Control*, 63(8):2508–2522, 2018.

[19] D. Dimarogonas, E. Frazzoli, and K. H. Johansson. Distributed event-triggered control for multi-agent systems. *IEEE Transactions on Automatic Control*, 57(5):1291–1297, 2012.

[20] W. B. Dunbar and R. M. Murray. Distributed receding horizon control for multi-vehicle formation stabilization. *Automatica*, 42(4):549–558, 2006.

[21] S. R. Etesami and T. Basar. Convergence time for unbiased quantized consensus over static and dynamic networks. *IEEE Transactions on Automatic Control*, 61(2):443–455, 2016.

[22] H. Farhangi. The path of the smart grid. *IEEE Power and Energy Magazine*, 8(1):18–28, 2010.

[23] 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 Transaction on Automatic Control*, 54(11):2560–2572, 2009.

[24] J. A. Garay. Reaching (and maintaining) agreement in the presence of mobile faults,. In *Proceedings of the 8th International Workshop on Distributed Algorithms, in: Lecture Notes in Computer Science*, volume 857, 1994.

[25] L. Guerrero-Bonilla, A. Prorok, and V. Kumar. Formations for resilient robot teams. *IEEE Robotics and Automation Letters*, 2(2):841–848, 2017.

[26] G. Guo, L. Ding, and Q. L. Han. A distributed event-triggered transmission strategy for sampled-data consensus of multi-agent systems. *Automatica*, 50:1489–1496, 2014.

[27] M. Heemels, K. H. Johansson, and P. Tabuada. An introduction to event-triggered and self-triggered control. In *Proc. 51th IEEE Conference on Decision and Control*, pages 3270–3285, 2012.

[28] F. Hohl. *Time Limited Blackbox Security: Protecting Mobile Agents From Malicious Hosts*. Springer, 1998.

[29] H. Ishii and R. Tempo. Distributed randomized algorithms for the PageRank computation. *IEEE Transactions on Automatic Control*, 55:1987–2002, 2010.

[30] H. Ishii and R. Tempo. The PageRank problem, multiagent consensus, and web aggregation: A systems and control viewpoint. *IEEE Control Systems Magazine*, 34(3):34–53, 2014.

[31] X. Jin. Fault tolerant finite-time leadercfollower formation control for autonomous surface vessels with LOS range and angle constraints. *Automatica*, 68:228–236, 2016.

[32] B. Johansson, A. Speranzon, M. Johansson, and K. H. Johansson. On decentralized negotiation of optimal consensus. *Automatica*, 44:1175–1179, 2008.

[33] Y. Kadowaki and H. Ishii. Event-based distributed clock synchronization for wireless sensor networks. *IEEE Transactions on Automatic Control*, 60:2266–2271, 2015.

[34] S. Kar and J. M. F. Moura. Distributed consensus algorithms in sensor networks: Quantized data and random link failures. *IEEE Transactions on Signal Processing*, 58(3):1383–1400, 2010.

[35] S. Karnouskos. Cyber-physical systems in the smartgrid. In *9th IEEE International Conference on Industrial Informatics*, 2011.

[36] A. Kashyap, T. Basar, and R. Srikant. Quantized consensus. *Automatica*, 43(7):1192–1203, 2007.

[37] T. Keviczky, F. Borrelli, and G. J. Balas. Decentralized receding horizon control for large scale dynamically decoupled systems. *Automatica*, 42(12):2105–2115, 2006.

[38] S. K. Khaitan, J. D. McCalley, and C. C. Liu. *Cyber Physical Systems Approach to Smart Electric Power Grid*. Springer, 2015.

[39] S. S. Kia, J. Cortes, and S. Martinez. Distributed event-triggered communication for dynamic average consensus in networked systems. *Automatica*, 59:112–119, 2015.

[40] R. M. Kieckhafer and M. H. Azadmanesh. Reaching approximate agreement with mixed-mode faults. *IEEE Transactions on Parallel and Distributed Systems*, 5(1):53–63, 1994.

[41] K. Kikuchi, A. Cetinkaya, T. Hayakawa, and H. Ishii. Stochastic communication protocols for multi-agent consensus under jamming attacks. In *Proc. 56th IEEE Conference on Decision and Control*, pages 1657–1662, 2017.

[42] Y. Kikuya, S. M. Dibaji, and H. Ishii. Fault tolerant clock synchronization over unreliable channels in wireless sensor nerworks. *IEEE Transactions on Control of Network Systems*, 5:1551–1562, 2018.

[43] J. Lavaei and R. M. Murray. Quantized consensus by means of gossip algorithm. *IEEE Transactions on Automatic Control*, 57(1):19–32, 2012.

[44] H. J. LeBlanc and X. Koutsoukos. Resilient first-order consensus and weakly stable, higher order synchronization of continuous-time networked multi-agent systems. *IEEE Transactions on Control of Network Systems*, 5(3):1219–1231, 2018.

[45] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram. Resilient asymptotic consensus in robust networks. *IEEE J. Selected Areas Comm.*, 31:766–781, 2013.

[46] 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*, 59(6):1673–1678, 2014.

[47] G. Loukas. *Cyber-Physical Attacks: A Growing Invisible Threat.* Elsevier, 2015.

[48] N. A. Lynch. *Distributed Algorithms.* Morgan Kaufmann, 1996.

[49] L. F. Ma, Z. D. Wang, and H. Lam. Eveny-triggered mean-square consensus control for tome-varying schochastic multi-agent system with sensor saturations. *IEEE Transactions on Automatic Control*, 62(7):3524–3531, 2016.

[50] S. Marti, T. J. Giuli, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, pages 255–265, 2000.

[51] N. Mathew, S. L. Smith, and S. L. Waslander. Multirobot rendezvous planning for recharging in persistent tasks. *IEEE Transactions on Robotics*, 31(1):128–142, 2015.

[52] H. Mendes, M. Herlihy, N. Vaidya, and V. K. Garg. Multidimensional agreement in Byzantine systems. *Distributed Computing*, 28(6):423–441, 2015.

[53] X. Meng and T. Chen. Event-based agreement protocols for multi-agent networks. *Automatica*, 49:2125–2132, 2013.

[54] X. Y. Meng, L. Xie, and Y. C. Soh. Asynchronous periodic event-triggered consensus for multi-agent systems. *Automatica*, 84:214–220, 2017.

[55] M. Mesbahi and M. Egerstedt. *Graph Theoretical Methods in Multiagent Networks*. Princeton Univ. Press, 2010.

[56] A. Mitra and S. Sundaram. Resilient distributed state estimation for LTI systems. *arXiv:1802.09651v1*, 2019.

[57] Y. Mo and B. Sinopoli. Secure control against replay attacks. In *Annual Allerton Conference on Communication, Control, and Computing*, pages 911–918, 2009.

[58] Y. Mo and B. Sinopoli. False data injection attacks in control systems. In *Preprints of the 1st workshop on Secure Control Systems*, pages 1–6, 2010.

[59] E. Montijano, E. Cristofalo, D. J. Zhou, M. Schwager, and C Sagués. Vision-based distributed formation control without an external positioning system. *IEEE Transactions on Robotics*, 32(2):339–351, 2016.

[60] M. A. Müller, M. Reble, and F. Allgöwer. Cooperative control of dynamically decoupled systems via distributed model predictive control. *International Journal of Robust and Nonlinear Control*, 22:1376–1397, 2012.

[61] H. Nishino and H. Ishii. Distributed detection of cyber attacks and faults for power systems. In *Proceedings of the 19th IFAC World Congress*, pages 11932–11937, 2014.

[62] P. Ogren, E. Fiorelli, and N. E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, 2004.

[63] H. Park and S. A. Hutchinson. Fault-tolerant rendezvous of multirobot systems. *IEEE Transactions on Robotics*, 33(3):565–582, 2017.

[64] F. Pasqualetti, A. Bicchi, and F. Bullo. Consensus computation in unreliable networks: A system theoretic approach. *IEEE Transactions on Automatic Control*, 57(1):90–104, 2012.

[65] C. D. Persis and P. Tesi. Input-to-state stabilizing control under denial-of-service. *IEEE Transactions on Automatic Control*, 60(11):2930–2944, 2015.

[66] P. Pierpaoli, D. Sauter, and M. Egerstedt. Fault tolerant control for networked mobile robots. In *IEEE Conference on Control Technology and Applications (CCTA)*, pages 374–379, 2018.

[67] W. Ren and R. W Beard. *Distributed Consensus in Multi-vehicle Cooperative Control*. Springer, 2008.

[68] A. Richards and J. P. How. Robust distributed model predictive control. *International Journal of Control*, 80(9):1517–1531, 2007.

[69] D. Saldaña, A. Prorok, S. Sundaram, M. F. M. Campos, and V. Kumar. Resilient consensus for time-varying networks of dynamic agents. In *Proc. American Control Conference*, pages 252–258, 2017.

[70] H. Sandberg, S. Amin, and K. H. (guest eds) Johansson. Special issue on cyberphycial security in networked control systems. *IEEE Control Systems Magazine*, 35(1), 2015.

[71] T. Sander and C. F. Tschudin. *Protecting Mobile Agents Against Malicious Hosts*. Springer, 1998.

[72] T. Sasaki, Y. Yamauchi, S. Kijima, and M. Yamashita. Mobile byzantine agreement on arbitrary network. In *Proceedings of the 17th International Conference on Principles of Distributed Systems, OPODIS13*, pages 236–250, 2013.

[73] L. Schenato and F. Fiorentin. Average timesynch: A consensus-based protocol for clock synchronization in wireless sensor networks. *Automatica*, 47(9):1878–1886, 2011.

[74] D. Senejohnny, P. Tesi, and C. D. Persis. A jamming-resilient algorithm for self-triggered network coordination. *IEEE Transactions on Control of Network Systems*, 5(3):981–990, 2018.

[75] G. Seyboth, D. V. Dimarogonas, and K. H. Johansson. Event-based broadcasting for multi-agent average consensus. *Automatica*, 49:245–252, 2013.

[76] I. Shames, A. Teixeira, H. Sandberg, and K. H. Johansson. Distributed fault detection for interconnected second-order systems. *Automatica*, 47(12):2757–2764, 2011.

[77] S. Sundaram and B. Gharesifard. Distributed optimization under adversarial nodes. *IEEE Transactions on Automatic Control*, 64(3):1063–1076, 2018.

[78] I. Suzuki and M. Yamashita. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM Journal on Computing*, 28(4):1347–1363, 1999.

[79] A. Teixeira, D. Pérez, H. Sandberg, and K. H. Johansson. Attack models and scenarios for networked control systems. In *Proceedings of the 1st Inter-*

national Conference on High Confidence Networked Systems, pages 55–64, 2012.

[80] R. Tempo and H. Ishii. Monte Carlo and Las Vegas randomized algorithms for systems and control. *European Journal of Control*, 13:189–203, 2007.

[81] J. Usevitch and D. Panagou. r-robustness and (r, s)-robustness of circulant graphs. In *Proc. 56th IEEE Conference on Decision and Control*, pages 4416–4421, 2017.

[82] N. H. Vaidya, L. Tsen, and G. Liang. Iterative approximate Byzantine cosnensus in arbitrary directed graphs. In *Proc. ACM Symp. on Principles of Distributed Computing*, pages 365–374, 2012.

[83] P. Velarde, J. M. Maestre, H. Ishii, and R. Negenborn. Vulnerabilities in Lagrange-based distributed model predictive control. *Optimal Control Applications and Methods*, 39:601–621, 2018.

[84] Y. Wang and H. Ishii. An event-triggered approach to quantized resilient consensus. In *Proc. European Control Conference*, pages 2719–2724, 2019.

[85] Y. Wang and H. Ishii. Resilient consensus through asynchronous event-based communication. In *Proc. American Control Conference*, pages 1842–1847, 2019.

[86] Y. Wang and H. Ishii. A distributed model predictive scheme for resilient consensus with input constraints. In *IEEE Conference on Control Technology and Applications (CCTA)*, to appear, 2019.

[87] Y. Wang and H. Ishii. Resilient consensus through event-based communication. *IEEE Transactions on Control of Network Systems*, to appear, 2019.

[88] F. Xiao and L. Wang. Asynchronous consensus in continous-time multi-agent systems with switching topology and time-varying delays. *IEEE Transactions on Automatic Control*, 53:1804–1816, 2008.

[89] A. J. Younge, G. Laszewski, L. Wang, S. Lopez-Alarcon, and W. Carithers. Efficient resource management for cloud computing environments. In *International Conference on Green Computing*, 2010.

[90] H. Yu and P. J. Antsaklis. Quantized output synchronization of network passive systems with event-driven communication. In *Proc. American Control Conference*, pages 5706–5711, 2012.

[91] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi. Internet of things for smart cities. *IEEE Internet of Things Journal*, 1(1):22–32, 2014.

[92] H. Zhang, E. Fata, and S. Sundaram. A notion of robustness in complex networks. *IEEE Transactions on Control of Network Systems*, 2(3):310–320, 2015.

[93] J. Zhao, O. Yagan, and V. Gligor. On connectivity and robustness in random intersection graphs. *IEEE Transactions on Automatic Control*, 62(5):2121–2136, 2017.

[94] Y. Zheng, S. E. Li, K. Q. Li, F. Borrelli, and J. K. Hedrick. Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies. *IEEE Transactions on Control Systems Technology*, 25(3):899–910, 2017.

[95] M. H. Zhu and S. Martínez. On resilient consensus against replay attacks in operator-vehicle networks. *Proc. American Control Conference*, pages 3553–3558, 2012.