Automatica 81 (2017) 123–132

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Resilient consensus of second-order agent networks: Asynchronous update rules with delays☆

Seyed Mehran Dibaji [a], Hideaki Ishii [b]

[a] Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
[b] Department of Computer Science, Tokyo Institute of Technology, Yokohama 226-8502, Japan

ARTICLE INFO

Article history:
Received 3 August 2015
Received in revised form
31 January 2017
Accepted 18 February 2017

Keywords:
Multi-agent systems
Cyber-security
Consensus problems

ABSTRACT

We study the problem of resilient consensus of sampled-data multi-agent networks with double-integrator dynamics. The term resilient points to algorithms considering the presence of attacks by faulty/malicious agents in the network. Each normal agent updates its state based on a predetermined control law using its neighbors' information which may be delayed while misbehaving agents make updates arbitrarily and might threaten the consensus within the network. Assuming that the maximum number of malicious agents in the system is known, we focus on algorithms where each normal agent ignores large and small state values among its neighbors to avoid being influenced by malicious agents. The malicious agents are assumed to be omniscient in that they know the updating times and delays and can collude with each other. We deal with both synchronous and partially asynchronous cases with delayed information and derive topological conditions in terms of graph robustness.

1. Introduction

In recent years, much attention has been devoted to the study of networked control systems with an emphasis on cyber security. Due to communications through shared networks, there are many vulnerabilities for potential attacks, which can result in irreparable damages. Conventional control approaches are often not applicable for resiliency against such unpredictable but probable misbehaviors in networks (e.g., Sandberg, Amin, & Johansson, 2015). One of the most essential problems in networked multi-agent systems is consensus where agents interact locally to achieve the global goal of reaching a common value. Having a wide variety of applications in UAV formations, sensor networks, power systems, and so on, consensus problems have been studied extensively (Mesbahi & Egerstedt, 2010; Ren & Cao, 2011). *Resilient consensus* points to the case where some agents in the network anonymously try to mislead the others or are subject to failures. Such malicious agents do not comply with the predefined interaction rule and might even prevent the normal agents from reaching consensus. This type of problems has a rich history in distributed algorithms in the area of computer science (see, e.g., Lynch, 1996) where the agents' values are often discrete and finite. It is interesting that randomization sometimes play a crucial role; see also Dibaji, Ishii, and Tempo (2016), Motwani and Raghavan (1995) and Tempo and Ishii (2007).

In such problems, the non-faulty agents cooperate by interacting locally with each other to achieve agreement. There are different techniques to mitigate the effects of attacks. In some solutions, each agent has a bank of observers to identify the faulty agents within the network using their past information. Such solutions are formulated as a kind of fault detection and isolation problems (Pasqualetti, Bicchi, & Bullo, 2012; Shames, Teixeira, Sandberg, & Johansson, 2011; Sundaram & Hadjicostis, 2011). However, identifying the malicious agents can be challenging and requires much information processing at the agents. In particular, these techniques usually necessitate each agent to know the topology of the entire network. This global information typically is not desirable in distributed algorithms. To overrule the effects of f malicious agents, the network has to be at least $(2f+1)$-connected.

There is another class of algorithms for resilient consensus where each normal agent disregards the most deviated agents in the updates. In this case, they simply neglect the information received from suspicious agents or those with unsafe values whether or not they are truly misbehaving. This class of algorithms

☆ This work was supported in part by the Japan Science and Technology Agency under the EMS-CREST program and by JSPS under Grant-in-Aid for Scientific Research Grant No. 15H04020. The material in this paper was presented at the 2015 American Control Conference, July 1–3, 2015, Chicago, IL, USA and at the 54th IEEE Conference on Decision and Control, December 15–18, 2015, Osaka, Japan. This paper was recommended for publication in revised form by Associate Editor Antonis Papachristodoulou under the direction of Editor Christos G. Cassandras.

E-mail addresses: dibaji@mit.edu (S.M. Dibaji), ishii@c.titech.ac.jp (H. Ishii).

http://dx.doi.org/10.1016/j.automatica.2017.03.008
0005-1098/

has been extensively used in computer science (Azevedo & Blough, 1998; Azadmanesh & Kieckhafer, 2002; Bouzid, Potop-Butucaru, & Tixeuil, 2010; Lynch, 1996; Plunkett & Fekete, 1998; Vaidya, Tseng, & Liang, 2012) as well as control (Dibaji & Ishii, 2015a,d; LeBlanc & Koutsoukos, 2012; LeBlanc, Zhang, Koutsoukos, & Sundaram, 2013); see also Feng, Hu, and Wen (2016) and Khanafer, Touri, and Başar (2012) for related problems. They are often called Mean Subsequence Reduced (MSR) algorithms, which was coined in Kieckhafer and Azadmanesh (1993). Until recently, this strategy had been studied mostly in the case where the agent networks form complete graphs. The authors of LeBlanc et al. (2013) have given a thorough study for the non-complete case and have shown that the traditional connectivity measure is not adequate for MSR-type algorithms to achieve resilient consensus. They then introduced a new notion called graph robustness. We note that most of these works have dealt with single-integrator and synchronous agent networks.

In this paper, we consider agents having second-order dynamics, which is a common model for autonomous mobile robots and vehicles. Such applications in fact provide motivations different from those in computer science as we will see. In our previous paper (Dibaji & Ishii, 2015a), an MSR-type algorithm has been applied to sampled-data second-order agent networks. We have considered the problem of resilient consensus when each agent is affected by at most f malicious agents among its neighbors. Such a model is called f-local malicious. We have established a sufficient condition on the underlying graph structure to reach consensus. It is stated in terms of graph robustness and is consistent with the result in LeBlanc et al. (2013) for the first-order agent case.

Here, the focus of our study is on the so-called f-total model, where the total number of faulty agents is at most f, which has been dealt with in, e.g., Azadmanesh and Kieckhafer (2002), Bouzid et al. (2010), Kieckhafer and Azadmanesh (1993), LeBlanc and Koutsoukos (2012), LeBlanc et al. (2013), Lynch (1996) and Vaidya et al. (2012). We derive a necessary and sufficient condition to achieve resilient consensus by an MSR-like algorithm. Again, we show that graph robustness in the network is the relevant notion. However, the f-total model assumes fewer malicious agents in the system, and hence, the condition will be shown to be less restrictive than that for the f-local case. The works (Bouzid et al., 2010; Kieckhafer & Azadmanesh, 1993; Lynch, 1996; Vaidya et al., 2012) have studied this model for the first-order agents case, but based on the Byzantine malicious agents, which are allowed to send different values to their neighbors. Such attacks may be impossible, e.g., if the measurements are made by on-board sensors in mobile robots.

Under the f-total model, we solve the resilient consensus problem using MSR-type algorithms for two different updating rules: Synchronous and partially asynchronous.[1] In the synchronous case, all agents simultaneously make updates at each time step using the current information of their neighbors. By contrast, in the asynchronous case, normal agents may decide to update only occasionally and moreover, the neighbors' data may be delayed. This is clearly a more vulnerable situation, allowing the adversaries to take advantage by quickly moving around. We consider the worst-case scenarios where the malicious agents are aware of the updating times and even the delays in the information of normal agents. The normal agents on the other hand are unaware of the updating times of their neighbors and hence cannot predict the plans of adversaries. For both cases, we develop graph robustness conditions for the overall network topologies. It will be shown that the synchronous updating rules require less connectivity than the asynchronous counterpart; see also Dibaji and Ishii (2015d) regarding corresponding results for first-order agent systems.

The main features of this work are three-fold: (i) We deal with second-order agents, which are more suitable for modeling networks of vehicles, but exhibit more complicated dynamics in comparison to the single-order case. (ii) For the malicious agents, we consider the f-total model, which is less stringent than the f-local case, but the analysis is more involved. (iii) In the asynchronous case with delayed information, we introduce a new update scheme, which is more natural in view of the current research in the area of multi-agent systems than those based on the so-called rounds, commonly employed in computer science as we discuss later.

The paper is organized as follows. Section 2 presents preliminaries for introducing the problem setting. Section 3 focuses on resilient consensus based on synchronous update rules. Section 4 is devoted to the problem of partial asynchrony with delayed information. We illustrate the results through a numerical example in Section 5. Finally, Section 6 concludes the paper. The material of this paper appears in Dibaji and Ishii (2015b,c) in preliminary forms; here, we present improved results with full proofs and more discussions. Further details can be found in Dibaji and Ishii (2017).

2. Problem setup

2.1. Graph theory notions

We recall some concepts on graphs (Mesbahi & Egerstedt, 2010). A directed graph (or digraph) with n nodes ($n > 1$) is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with the node set $\mathcal{V} = \{1, \ldots, n\}$ and the edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The edge $(j, i) \in \mathcal{E}$ means that node i has access to the information of node j. If $\mathcal{E} = \{(i, j) : i, j \in \mathcal{V},\ i \neq j\}$, the graph is said to be complete. For node i, the set of its neighbors, denoted by $\mathcal{N}_i = \{j : (j, i) \in \mathcal{E}\}$, consists of all nodes having directed edges toward i. The degree of node i is the number of its neighbors and is denoted by $d_i = |\mathcal{N}_i|$. The adjacency matrix $A = [a_{ij}]$ is given by $a_{ij} \in [\gamma, 1)$ if $(j, i) \in \mathcal{E}$ and otherwise $a_{ij} = 0$, where $\gamma > 0$ is a fixed lower bound. We assume that $\sum_{j=1, j\neq i}^{n} a_{ij} \leq 1$. Let $L = [l_{ij}]$ be the Laplacian matrix of $\mathcal{G}$, whose entries are defined as $l_{ii} = \sum_{j=1, j\neq i}^{n} a_{ij}$ and $l_{ij} = -a_{ij}$, $i \neq j$; we can see that the sum of the elements of each row of L is zero.

A path from node v_1 to v_p is a sequence $(v_1, v_2, \ldots, v_p)$ in which $(v_i, v_{i+1}) \in \mathcal{E}$ for $i = 1, \ldots, p-1$. If there is a path between each pair of nodes, the graph is said to be strongly connected. A directed graph is said to have a directed spanning tree if there is a node from which there is a path to every other node in the graph.

For the MSR-type resilient consensus algorithms, the critical topological notion is graph robustness, which is a connectivity measure of graphs. Robust graphs were introduced in LeBlanc et al. (2013) for the analysis of resilient consensus of first-order multi-agent systems.

Definition 2.1. The digraph $\mathcal{G}$ is (r, s)-robust ($r, s < n$) if for every pair of nonempty disjoint subsets $\mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{V}$, at least one of the following conditions is satisfied:

$$1.\ \mathcal{X}_{\mathcal{S}_1}^r = \mathcal{S}_1, \qquad 2.\ \mathcal{X}_{\mathcal{S}_2}^r = \mathcal{S}_2, \qquad 3.\ |\mathcal{X}_{\mathcal{S}_1}^r| + |\mathcal{X}_{\mathcal{S}_2}^r| \geq s,$$

where $\mathcal{X}_{\mathcal{S}_\ell}^r$ is the set of all nodes in $\mathcal{S}_\ell$ which have at least r incoming edges from outside of $\mathcal{S}_\ell$. In particular, graphs which are $(r, 1)$-robust are called r-robust.

The following lemma helps to have a better understanding of (r, s)-robust graphs (LeBlanc, 2012).

Lemma 2.2. *For an (r, s)-robust graph $\mathcal{G}$, the following hold:*

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

[1] The term *partially* asynchronous refers to the case where agents share some level of synchrony by having the same sampling times; however, they make updates at different times based on delayed information (Bertsekas & Tsitsiklis, 1989). This is in contrast to the *fully* asynchronous case where agents can be facilitated with their own clocks; such settings are studied in, e.g., Qin, Yu, and Hirche (2012).