Systems & Control Letters 79 (2015) 23–29

Contents lists available at ScienceDirect

Systems & Control Letters

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

# Consensus of second-order multi-agent systems in the presence of locally bounded faults☆

Seyed Mehran Dibaji, Hideaki Ishii *

*Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, 4259-J2-54, Nagatsuta-cho, Midori-ku Yokohama 226-8502, Japan*

ARTICLE INFO

*Article history:*
Received 21 November 2014
Received in revised form
11 February 2015
Accepted 23 February 2015
Available online 27 March 2015

*Keywords:*
Multi-agent systems
Cyber-security
Consensus

ABSTRACT

We propose an algorithm for consensus of second-order sampled-data multi-agent systems in the presence of misbehaving agents. Each normal agent updates its states following a predetermined control law based on local information while some malicious agents make updates arbitrarily. The normal agents do not know the global topology of the network, but have prior knowledge on the maximum number of malicious agents in their neighborhood. Under the assumption that the network has sufficient connectivity in terms of robustness, we develop a resilient algorithm where each agent ignores the neighbors which have large and small position values to avoid being influenced by malicious agents.



## 1. Introduction

Recently, in the area of networked control systems, consideration of cyber security has become important since such systems are nowadays often connected to general purpose networks, e.g., the Internet and wireless communication. Malicious attacks can lead the systems to hazardous operations and might cause physical faults or even accidents. Safe distributed algorithms in the presence of faulty behaviors and adversarial agents have been widely studied in computer science [1,2] and control [3–5].

In this paper, we consider networks of agents which interact with each other to accomplish a global objective. In such systems, malicious intruders may take control of some agents and influence other agents to keep them from completing their planned tasks without being noticed. Here, we consider consensus, one of the basic problems in multi-agent systems, where the objective is agreement on some state values among the agents [6].

Resilient algorithms for multi-agent systems have appeared in the literature and can be classified into two approaches. One is to achieve consensus among the non-faulty agents by detecting and isolating malicious agents in the network. In the works of [7,8], techniques of observers for systems with unknown inputs are developed for a consensus problem. The papers [4,5] also deal with observer-based methods for fault detection when the agents have second-order dynamics. The other approach aims at consensus by simply ignoring suspicious agents whether or not they are truly faulty. The paper [9] first proposed a consensus algorithm with this idea. The network considered there is however a complete graph. Afterwards, there are some works [10,11] which studied the algorithm for partially connected networks. The term mean subsequence reduced (MSR) algorithms coined by [12] for this family of algorithms has been used in literature (e.g., [13,14]). On the other hand, the papers [14,15] introduce a novel notion of graph robustness to characterize the necessary network structure; other related works include [16,17]. It is noted that these works study only agents whose dynamics is represented as a single integrator.

Here, we focus on resilient consensus of sampled-data double-integrator multi-agent systems in the presence of locally bounded malicious agents. Consensus problems for second-order agent dynamics are motivated by vehicle applications and have been studied, e.g., in [18,19]. Following the second approach mentioned above, we propose a new algorithm to tackle the problem. The difficulty of this problem is two-fold: (i) The presence of malicious agents which might try to deviate the network not to reach consensus and (ii) more complicated dynamics due to the double-integrator agents, which requires agreement in both position and velocity values. In our strategy, the non-faulty normal agents are equipped with an algorithm to collect the neighbors' positions, but to ignore a certain number of them. Specifically, in their updates, they leave out those that take large and small values. In this way, these agents can avoid being affected by the suspicious ones in the course of arriving at consensus. We show that the notion of graph

☆ This work was supported by Japan Science and Technology Agency under the EMS-CREST program.

* Corresponding author.
*E-mail addresses:* dibaji@sc.dis.titech.ac.jp (S.M. Dibaji), ishii@dis.titech.ac.jp (H. Ishii).

http://dx.doi.org/10.1016/j.sysconle.2015.02.005
0167-6911/

robustness from [14,15] indeed plays an important role to guarantee sufficient connectivity among the agents.

The outline of this paper is as follows. In Section 2, we present preliminary material and then formulate the resilient consensus problem. In Section 3, the proposed MSR-based algorithm is presented. In Section 4, a sufficient condition and a necessary condition are derived on the network topology for resilient consensus. In Section 5, we illustrate the effectiveness of the algorithm through a numerical example. Finally, Section 6 concludes the paper. This paper is based on the conference version [20] and contains extended results with their full proofs.

## 2. Problem formulation

In this section, we first provide some notations related to graphs and then introduce the problem setting.

### 2.1. Graph related notions

Given a network of $n$ agents ($n > 1$), we use a directed graph (or a digraph) $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to model the interaction network among agents, where $\mathcal{V} = \{1, \dots, n\}$ denotes the node set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set. The edge $(j, i) \in \mathcal{E}$ indicates that node $i$ can receive information from node $j$. Such edge is called an incoming edge of node $i$. If $\mathcal{E} = \mathcal{V} \times \mathcal{V}$, the corresponding graph is called a complete graph. For node $i$, the set of neighbors is given by $\mathcal{N}_i = \{j : (j, i) \in \mathcal{E}\}$, and its degree is denoted by $d_i = |\mathcal{N}_i|$.

The adjacency matrix $A$ is given by $a_{ij} \in (\gamma, 1)$ if $(j, i) \in \mathcal{E}$ and otherwise $a_{ij} = 0$ with $\sum_{j=1}^{n} a_{ij} \leq 1$, where $\gamma > 0$ is a fixed lower bound. Then, the Laplacian matrix $L$ is defined by $l_{ii} = \sum_{j=1, j\neq i}^{n} a_{ij}$ and $l_{ij} = -a_{ij}$, $i \neq j$. It is clear that the sum of the elements of each row in the Laplacian matrix is zero.

A path from node $v_1$ to $v_p$ is a sequence $(v_1, v_2, \dots, v_p)$ such that $(v_i, v_{i+1}) \in \mathcal{E}$ for $i = 1, \dots, p-1$. In the directed graph $\mathcal{G}$, if there is a path between each pair of nodes the graph is said to be strongly connected. The connectivity (or the vertex connectivity) $K(\mathcal{G})$ of the graph $\mathcal{G}$ is the minimum number of vertices such that the graph formed by removing the vertices and the all edges associated with the vertices is not strongly connected. The graph is said to be $\kappa$-connected if $K(\mathcal{G}) \geq \kappa$.

We employ the notion of robustness, which is a connectivity measure for graphs. Such connectivity was introduced in [14] for analysis of resilient consensus of first-order multi-agent systems in the $f$-local malicious models.

**Definition 2.1.** The digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is $(r, s)$-robust ($r, s < n$) if for every pair of nonempty disjoint subsets $\mathcal{V}_1, \mathcal{V}_2 \subset \mathcal{V}$, at least one of the following conditions holds: (i) $|\mathcal{X}_{\mathcal{V}_1}^r| = |\mathcal{V}_1|$, (ii) $|\mathcal{X}_{\mathcal{V}_2}^r| = |\mathcal{V}_2|$, (iii) $|\mathcal{X}_{\mathcal{V}_1}^r| + |\mathcal{X}_{\mathcal{V}_2}^r| \geq s$, where $\mathcal{X}_{\mathcal{V}_\ell}^r$ is the set of all nodes in $\mathcal{V}_\ell$ which have at least $r$ incoming edges from outside of $\mathcal{V}_\ell$. In particular, graphs which are $(r, 1)$-robust are called $r$-robust.

The next lemma shows some properties of robust graphs.

**Lemma 2.2** ([21]). *Suppose $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is an $r$-robust network ($r \geq 1$). Then it has the following properties:*

(i) *$r \leq \lceil n/2 \rceil$. Also, if $\mathcal{G}$ is a complete graph, then it is $r'$-robust for all $0 < r' \leq \lceil n/2 \rceil$.*
(ii) *$\mathcal{G}$ is at least $r$-connected.*
(iii) *Any subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ of $\mathcal{G}$ where at most $q$ incoming edges to each node has been removed is $(r - q)$-robust.*
(iv) *$\mathcal{G}$ has a directed spanning tree.*
(v) *The graph $\mathcal{G}' = (\mathcal{V} \cup \{v_0\}, \mathcal{E} \cup \mathcal{E}_0)$, where $v_0$ is a vertex added to $\mathcal{G}$ and $\mathcal{E}_0$ is the edge set related to $v_0$, is $r$-robust if $d_{v_0} \geq r$.*

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

By (ii) of the lemma, we see that robustness is stronger than the common metric of connectivity. Fig. 1 gives an example of a 3-robust graph with seven nodes. Its robustness can directly be checked based on Definition 2.1.

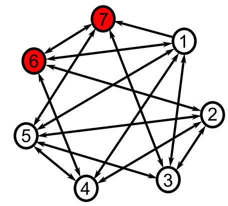


**Fig. 1.** A 3-robust digraph with seven nodes.

### 2.2. Second-order consensus protocol

Consider a network represented by the digraph $\mathcal{G}$. Each agent $i$ has a double-integrator dynamics given by

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

where $r_i(t) \in \mathbb{R}$ and $v_i(t) \in \mathbb{R}$ are the position and the velocity of the $i$th agent at time $t \geq 0$, respectively, and $u_i(t)$ is the control input applied to the agent. We study the discretized version of the system [18]. After discretization with sampling period $T > 0$, the system (1) becomes

$$\begin{aligned} r_i[k+1] &= r_i[k] + T v_i[k] + \frac{T^2}{2} u_i[k], \\ v_i[k+1] &= v_i[k] + T u_i[k], \quad i = 1, \dots, n, \end{aligned} \tag{2}$$

where $r_i[k]$, $v_i[k]$, and $u_i[k]$ are, respectively, the position, the velocity, and the control of the $i$th agent at $t = kT$.

At each discrete time $k$, the agents update their positions and velocities based upon the current topology of the graph $\mathcal{G}[k]$, which is a subgraph of $\mathcal{G}$ and is specified later. The objective of the networked agents is consensus in the sense that they come to formation and then stop asymptotically: $r_i[k] - r_j[k] \to \Delta_{ij}$, $v_i[k] \to 0$ as $k \to \infty$. Here, we have $\Delta_{ij} = \delta_i - \delta_j$, where $\delta_i \in \mathbb{R}$ represents the desired relative position of agent $i$ in a formation.

For each agent, the control law is based on the relative positions with its neighbors and its own velocity:

$$u_i[k] = -\sum_{j=1}^{n} a_{ij}[k]\left[(r_i[k] - \delta_i) - (r_j[k] - \delta_j)\right] - \alpha v_i[k], \tag{3}$$

where $a_{ij}[k]$ is the $(i, j)$ entry of the adjacency matrix $A[k] \in \mathbb{R}^{n\times n}$ corresponding to $\mathcal{G}[k]$ and $\alpha$ is a positive scalar. In [18], it is shown that consensus can be obtained under this control by properly choosing $\alpha$, $T$, and $\eta$, where $\eta$ is a parameter such that for each $k_0 \geq 0$, the union of $\mathcal{G}[k]$ across $k \in [k_0, k_0 + \eta]$ has a directed spanning tree.

In the real world, however, when some agents fail or are attacked, they may not follow the pre-defined control (3). In the next subsection, we introduce necessary definitions and then formulate the resilient consensus problem in the presence of malicious agents.

Finally, we represent the networked system in a vector form. Let $\hat{r}_i[k] = r_i[k] - \delta_i$, $\hat{r}[k] = \left[\hat{r}_1[k] \ \cdots \ \hat{r}_n[k]\right]^T$, $v[k] = [v_1[k] \ \cdots \ v_n[k]]^T$, and $u[k] = [u_1[k] \ \cdots \ u_n[k]]^T$. Then, the system in (2) is expressed as

$$\begin{aligned} \hat{r}[k+1] &= \hat{r}[k] + T v[k] + \frac{T^2}{2} u[k], \\ v[k+1] &= v[k] + T u[k], \end{aligned} \tag{4}$$

and the control law (3) is written as

$$u[k] = -L[k]\hat{r}[k] - \alpha v[k], \tag{5}$$

where $L[k]$ is the Laplacian matrix for the graph $\mathcal{G}[k]$.