Finite-time Consensus in the Presence of Malicious Agents

Extended Abstract

Sachit Rao
IIIT Bangalore
Bangalore, India
sachit@iiitb.ac.in

Shrisha Rao
IIIT Bangalore
Bangalore, India
shrao@ieee.org

ABSTRACT

A finite-time consensus protocol is developed for a connected network of agents, where communication between agents occurs locally, some of the agents are malicious, and the non-malicious or cooperative agents do not know the identities of the malicious ones. The agents are modeled with first-order dynamics, and the inputs to each agent that enable consensus are designed using the principles of sliding mode control (SMC). The use of the SMC algorithm guarantees finite-time consensus and ensures that in the transient stage, the agents' states are contained within the convex hull formed by their initial conditions (ICs). With this feature and by modeling the network as a connected graph, the protocol guarantees consensus amongst the cooperative agents when the malicious agents transmit values of their states lying outside the convex hull of ICs, and the graph formed by the cooperative agents with the removal of the malicious agents is strongly connected. The protocol does not require a cooperative agent to know the number of malicious or other cooperative agents in the network, and is based only on local communication.

KEYWORDS

consensus; sliding mode control; convex hull; guaranteed convergence

ACM Reference Format:
Sachit Rao and Shrisha Rao. 2021. Finite-time Consensus in the Presence of Malicious Agents: Extended Abstract. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 3 pages.

1 INTRODUCTION

Consensus algorithms for a connected network of agents, when some of the agents in the network become malicious, for instance, as the result of an external attack, are of current interest. Such algorithms, which are distributed in nature and are based on local information exchanges, are denoted as *resilient* consensus algorithms or protocols (RCP) [14, 20]. The malicious agents (MA) can drive the consensus value of the cooperative (CO) agents to an undesirable value or even to an unsafe region.

In this work, an RCP is developed using concepts from SMC theory, for a network of agents defined by first-order dynamics [18]; there are recent examples of such protocols [3, 6, 8, 13]. The input to the dynamic system is chosen to be discontinuous on a switching manifold. As a result, "sliding mode", which defines the closed-loop dynamics, occurs within a finite-time interval (see Utkin [17] for properties of sliding mode). Due to this property, an SMC-based RCP can be designed to lead to consensus within a finite-time interval—this is in contrast with most non-SMC based protocols, where only asymptotic consensus is guaranteed.

In the distributed computing literature [2, 12], consensus is described as a decision problem for n processes, some of which may be faulty [10]. The consensus protocol must fulfill the following conditions: *i.* Consistency: all non-faulty processes must return the same output value; *ii.* Termination: every non-faulty process must return an output value within some finite time; and *iii.* Validity: if every process is given the same input value, then every process must return that very value as output. Certain impossibility results follow, of which the best known is that in a strictly asynchronous system, consensus cannot be achieved even if a single processor fails [7], though this can be relaxed if even weak synchrony is possible [5].

An RCP is similarly expected to satisfy the conditions that define consensus amongst CO agents, in the presence of MAs. These conditions are the *agreement* condition—the states of any connected pair of CO agents should reach the same value; and the *validity* condition—the trajectories of the CO agents should lie within the interval defined by the agents' ICs. Most RCPs [1, 4, 9, 11, 16, 19, 20], require CO agents to know the number of MAs in the network.

With our SMC-based RCP described here, this requirement is eliminated for MAs that send values that lie outside the convex hull formed by the agents' ICs. An MA can also send different values, that lie outside the convex hull, to different CO agents. This is an assumption made for Fault Identification and Detection algorithms—a faulty agent can be detected only if its input drives its state out of some known bounds [15, 20]. The SMC-based RCP ensures *i.* that the CO agents satisfy the validity and the agreement conditions if and only if the sub-graph induced by the removal of the MAs is connected; and *ii.* consensus occurs amongst the CO agents for any attack model and network topology, for instance, one with cycles and cliques (see LeBlanc *et al.* [11] for types of attack models in networks with MAs).

2 MAIN RESULTS

The SMC-based RCP is developed for a network of agents as shown in Fig. 1, where some of the agents are MAs. The network topology is defined by an undirected graph G with n vertices. This assumption implies that information flow is in both directions between a pair of connected agents. For G, the symmetric Laplacian matrix $\mathbf{L}(G) \in \mathfrak{R}^{n\times n}$ which satisfies $\mathbf{L}(G) = \mathbf{D}(G) - \mathbf{A}(G)$ can be defined. $\mathbf{D}(G)$ is a diagonal matrix consisting of the positive integers d_{ii}, $1 \leq i \leq n$ that denote the number of neighbors of agent i; $\mathbf{A}(G)$ is also a

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

symmetric matrix with elements $a_{ij} = a_{ji} = +1$ if agents i and j are connected to each other, and 0 otherwise. For the undirected graph G, these integers satisfy $\sum a_{ij} = d_i$. Since the rank of $\mathbf{L}(G)$ is $(n-1)$, a vector $\mathbf{x}_r \in \Re^{n\times 1}$, with non-zero identical elements, can be found such that $\mathbf{L}(G)\mathbf{x}_r = \mathbf{0}$. It is this property of the Laplacian matrix that is used to provide consensus.

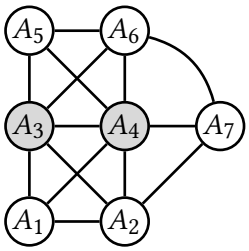

Figure 1: Network of CO (in white) and MA (in gray) agents

The agents in the network are defined by the dynamics $\dot{x}_i = u_i$, $1 \le i \le n$, $x_i(t=0) \ne x_j(t=0)$, $i \ne j$. Thus, the SMC-based RCP involves the selection of the inputs u_i such that within a finite-time interval T_C, the states satisfy $x_i(T_C) = x_j(T_C)$, $i \ne j$, and $\forall\, t \ge 0$, $x_i(t) \in \text{Conv}(x_i(t=0),\ 1 \le i \le n)$, where $\text{Conv}(x_i(t=0),\ 1 \le i \le n)$ is the convex hull formed by the agents' ICs. CO agents in the network apply the same input form, while an MA, with state x_{Mj}, is one that does not apply the same form of the input as the CO agents *and* also transmits as its state information, a value that satisfies $x_{Mj}(t) \notin \text{Conv}(x_i(t=0),\ 1 \le i \le n)$ to its neighbors. Note that an MA can also send different values to different neighbors and CO agents can identify and ignore information from MAs. With these definitions, the main results are presented as follows.

Theorem 2.1. *For a network comprised of f MAs and $(n-f)$ CO agents, the SMC protocol*

$$u_i = -M\,sign(s_i),\ s_i = \mathbf{L_i}(G_R)\mathbf{x}_{Ci},\ M > 0, \tag{1}$$

leads to consensus amongst the CO agents if and only if the graph G_R formed by the removal of the MAs is connected, where $\mathbf{L_i}(G_R)$ is the row i of the Laplacian matrix of the reduced graph G_R, and $\mathbf{x}_{Ci}$ is the vector of the CO agents' states.

Proof. If $f = 0$, then it is known [13] that the consensus protocol (1) leads to consensus amongst the agents within a finite-time interval. The proof [13] is extended when the network has MAs. When $0 < f < n$, a CO agent can disregard the information sent by MAs connected to it. Let CO agent i be connected to $0 \le f_i < d_{ii}$ MAs. Since agent i disregards information from the f_i MAs, the elements of row i of the Laplacian matrix from the original graph G become $d_{iiR} = (d_{ii} - f_i) > 0$, and $a_{ij} = 0$, where agent j is MA.

Similarly, let all CO agents disregard information from MAs. Now, if graph G_R consisting of only CO agents is connected, then its Laplacian matrix $\mathbf{L}(G_R)$ is also rank deficient $(n-f-1)$, and satisfies all properties of any graph Laplacian matrix. Now, following the earlier proof [13], for the graph with only CO agents, the SMC-based RCP (1) guarantees consensus within a finite-time interval. □

Remarks: **1.** The consensus value is the average of the minimum and maximum ICs of the CO agents; **2.** the consensus time can be tuned using the control gain M; **3.** the RCP (1) guarantees consensus amongst CO agents for the case when MAs transmit the same state to all their neighbors that also satisfy $x_{Mj}(t) \in \text{Conv}(x_i(t=0),\ 1 \le i \le n)$, in this case, the consensus value is decided by the values x_{Mj}.

3 SIMULATION RESULTS

Agents A_3 and A_4 in Fig. 1 are MAs; the graph with the removal of these is connected. The ICs of the CO agents are randomly chosen between [0, 1]; the control gain is set to $M = 5$. Fig, 2 shows the CO agents continuing to be in consensus when $A_{3,4}$ become malicious after they reach a consensus with their neighbors. The result where the CO agents reach consensus at the value of the single MA that lies within the convex hull of ICs is validated in Fig. 3. As can be seen, this result satisfies both the agreement and validity conditions required in an RCP.

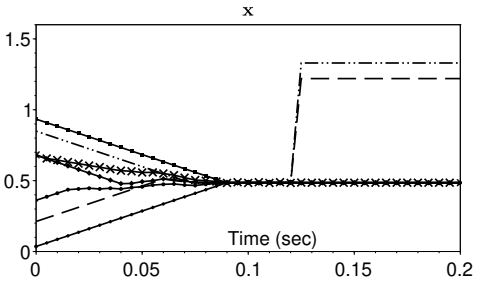

Figure 2: Agents "become" malicious for $t > T_C$

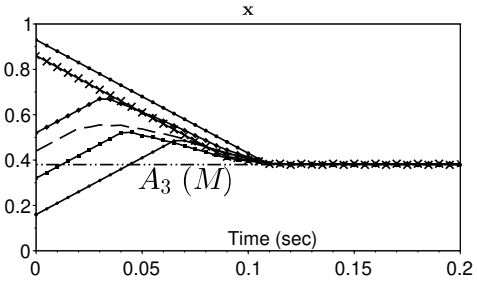

Figure 3: Single MA with state $\in$ Conv$(x_i(t=0))$

4 CONCLUSIONS

The SMC-based RCP given here has a unified feature and leads to finite-time consensus for a variety of MAs and network topologies without needing knowledge of the MAs. Consensus can be guaranteed amongst the CO agents even if some agents become malicious after a time interval. Since the protocol does not require heavy computation by each agent, it is simple to implement and can be used in diverse applications.

REFERENCES

[1] Waseem Abbas, Yevgeniy Vorobeychik, and Xenofon Koutsoukos. 2014. Resilient consensus protocol in the presence of trusted nodes. In *2014 7th International Symposium on Resilient Control Systems (ISRCS)*. Denver, CO. https://doi.org/10.1109/ISRCS.2014.6900100

[2] Hagit Attiya and Jennifer Welch. 2004. *Distributed Computing: Fundamentals, Simulations, and Advanced Topics* (2 ed.). Wiley-Interscience.

[3] Jorge Dávila. 2014. Distributed tracking of first order systems using second-order sliding modes. *IFAC Proceedings Volumes* 47, 3 (2014), 1392–1397. https://doi.org/10.3182/20140824-6-ZA-1003.00816 19th IFAC World Congress.

[4] Seyed Mehran Dibaji, Hideaki Ishii, and Roberto Tempo. 2018. Resilient Randomized Quantized Consensus. *IEEE Trans. Autom. Control* 63, 8 (2018), 2508–2522. https://doi.org/10.1109/TAC.2017.2771363

[5] Cynthia Dwork, Nancy A. Lynch, and Larry Stockmeyer. 1988. Consensus in the Presence of Partial Synchrony. *J. ACM* 35, 2 (April 1988), 288–323. https://doi.org/10.1145/42282.42283

[6] Antonella Ferrara and Massimo Zambelli. 2019. Integral Second-Order Sliding Modes for Robust Prescribed-Time Leader-Follower Consensus Control with Partial Information. In *2019 IEEE 58th Conference on Decision and Control (CDC)*. Nice, France, 7863–7868. https://doi.org/10.1109/CDC40024.2019.9029256

[7] Michael J. Fischer, Nancy A. Lynch, and Michael S. Paterson. 1985. Impossibility of distributed consensus with one faulty process. *J. ACM* 32, 2 (April 1985), 374–382. https://doi.org/10.1145/3149.214121

[8] Mauro Franceschelli, Alessandro Giua, and Alessandro Pisano. 2017. Finite-Time Consensus on the Median Value With Robustness Properties. *IEEE Trans. Autom. Control* 62, 4 (April 2017), 1652–1667. https://doi.org/10.1109/TAC.2016.2590602

[9] Jinbo Huang, Yiming Wu, Liping Chang, Meiling Tao, and Xiongxiong He. 2019. Resilient consensus with switching networks and heterogeneous agents. *Neurocomputing* 341 (2019), 70–79. https://doi.org/10.1016/j.neucom.2019.03.018

[10] Pankaj Jalote. 1994. *Fault Tolerance in Distributed Systems.* Prentice-Hall, Inc.

[11] Heath J. LeBlanc, Haotian Zhang, Xenofon Koutsoukos, and Shreyas Sundaram. 2013. Resilient Asymptotic Consensus in Robust Networks. *IEEE J. Sel. Areas Commun.* 31, 4 (April 2013), 766–781. https://doi.org/10.1109/JSAC.2013.130413

[12] Nancy A. Lynch. 1996. *Distributed Algorithms.* Morgan Kaufmann.

[13] Sachit Rao and Debasish Ghose. 2011. Sliding mode control-based algorithms for consensus in connected swarms. *Internat. J. Control* 84, 9 (2011), 1477–1490. https://doi.org/10.1080/00207179.2011.602834

[14] Yilun Shang. 2018. Resilient consensus of switched multi-agent systems. *Systems & Control Letters* 122 (Dec. 2018), 12–18. https://doi.org/10.1016/j.sysconle.2018.10.001

[15] Daniel Silvestre, Paulo Rosa, Jo ao P. Hespanha, and Carlos Silvestre. 2014. Finite-time average consensus in a Byzantine environment using Set-Valued Observers. In *2014 American Control Conference*. 3023–3028. https://doi.org/10.1109/ACC.2014.6859426

[16] Shreyas Sundaram and Bahman Gharesifard. 2015. Consensus-based distributed optimization with malicious nodes. In *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. 244–249. https://doi.org/10.1109/ALLERTON.2015.7447011

[17] Vadim I. Utkin. 1992. *Sliding modes in control and optimization.* Springer-Verlag.

[18] Jan C. Willems. 1991. Paradigms and Puzzles in the Theory of Dynamical Systems. *IEEE Trans. Autom. Control* 36, 3 (March 1991). https://doi.org/10.1109/9.73561

[19] Yiming Wu and Xiongxiong He. 2020. Finite-Time Consensus-Based Clock Synchronization Under Deception Attacks. *IEEE Access* 8 (2020), 110748–110758. https://doi.org/10.1109/ACCESS.2020.3002577

[20] Yiming Wu, Ming Xu, Ning Zheng, and Xiongxiong He. 2020. Event-Triggered Resilient Consensus for Multi-Agent Networks Under Deception Attacks. *IEEE Access* 8 (2020), 78121–78129. https://doi.org/10.1109/ACCESS.2020.2989743