

Consensus of Multi-Agent Networks in the Presence of Adversaries Using Only Local Information

Heath J. LeBlanc
Institute for Software
Integrated Systems
Dept. of Electrical Engineering
and Computer Science
Vanderbilt University
Nashville, TN, USA
heath.j.leblanc@vanderbilt.edu

Haotian Zhang
Department of Electrical and
Computer Engineering
University of Waterloo
Waterloo, ON, Canada
h223zhan@uwaterloo.ca

Shreyas Sundaram
Department of Electrical and
Computer Engineering
University of Waterloo
Waterloo, ON, Canada
ssundara@uwaterloo.ca

Xenofon Koutsoukos
Institute for Software
Integrated Systems
Dept. of Electrical Engineering
and Computer Science
Vanderbilt University
Nashville, TN, USA
xenofon.koutsoukos@vanderbilt.edu

ABSTRACT

This paper addresses the problem of resilient consensus in the presence of misbehaving nodes. Although it is typical to assume knowledge of at least some nonlocal information when studying secure and fault-tolerant consensus algorithms, this assumption is not suitable for large-scale dynamic networks. To remedy this, we emphasize the use of local strategies to deal with resilience to security breaches. We study a consensus protocol that uses only local information and we consider worst-case security breaches, where the compromised nodes have full knowledge of the network and the intentions of the other nodes. We provide necessary and sufficient conditions for the normal nodes to reach consensus despite the influence of the malicious nodes under different threat assumptions. These conditions are stated in terms of a novel graph-theoretic property referred to as *network robustness*.

Categories and Subject Descriptors

C.2.4 [Computer-Communication Networks]: Distributed Systems; H.1.1 [Models and Principles]: Systems and Information Theory—General Systems Theory

General Terms

Algorithms, Security, Theory

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

HiCoNS'12, April 17–18, 2012, Beijing, China.

Copyright 2012 ACM 978-1-4503-1263-9/12/04 ...\$10.00.

Keywords

Consensus, Multi-agent network, Resilience, Adversary

1. INTRODUCTION

The engineering community has witnessed a paradigm shift from centralized to distributed system design, propelled by advances in networking and low-cost, high performance embedded systems. In particular, this has led to significant interest in the design and analysis of *multi-agent networks*. A multi-agent network consists of a set of individuals called *agents*, or *nodes*, equipped with some means of sensing or communicating along with computational resources and possibly actuation. Through a medium, which is referred to as the *network*, the agents share information in order to achieve specific *group objectives*. Some examples of group objectives include consensus [22, 26], synchronization [6, 27], surveillance [5], and formation control [9]. In order for the group objectives to be achieved, *distributed algorithms* are used to coordinate the behavior of the agents.

There are several advantages to using multiple agents over a single one. First, the objective may be complex and challenging, or possibly even infeasible for a single agent to achieve. Second, employing many agents can provide robustness in the case of failures or faults. Third, networked multi-agent systems are flexible and can support reconfigurability. Finally, there are performance advantages that can be leveraged from multiple agents. For example, in surveillance and monitoring applications, a multi-agent network provides redundancy and increased fidelity of information [5, 14].

Along with the advantages come certain challenges. Perhaps the most fundamental challenge in the design of networked multi-agent systems is the restriction that the coordination algorithms use only *local information*, i.e., information obtained by the individual agent through sensor measurements, calculations, or communication with neighbors in the network. In this manner, the feedback control laws must be *distributed*.

A second challenge lies in the fact that not only is each agent typ-

ically a dynamical system, but the network itself is dynamic. This challenge arises because the agents may be mobile and the environment may be changing, thus giving rise to dynamic (or switching) networks. Since the distributed algorithms depend directly on the network, this additional source of dynamics can affect the stability and performance of the networked system.

An especially important challenge is that multi-agent networks, like all large-scale distributed systems, have many entry points for malicious attacks or intrusions. For the success of the group objective, it is important that the cooperative control algorithms are designed in such a way that they can withstand the compromise of a subset of the nodes and *still guarantee some notion of correct behavior at a minimum level of performance*. We refer to such a multi-agent network as being *resilient* to adversaries. Given the growing threat of malicious attacks in large-scale cyber-physical systems, this is an important and challenging problem [4].

One of the most fundamental group objectives is to reach consensus on a quantity of interest. This concept is deeply intuitive, yet imprecise. Hence, there are several variations on how consensus problems are defined. At one extreme, consensus may be *unconstrained*, and there is no restriction on the agreement quantity. In other cases, consensus may be *partially constrained* by some rule or prescribed to lie in a set of possible agreement values which are in some way reasonable to the problem at hand. At the other extreme, consensus may be *function constrained*, or χ -constrained, in which case the consensus value must satisfy a particular function of the initial values of the nodes [7, 28]. In all of these cases, it is important that consensus algorithms be *resilient* to various forms of uncertainty, whether the source of uncertainty is caused by implementation effects, faults, or security breaches.

The problem of reaching consensus resiliently in the presence of misbehaving nodes has been studied in distributed computing [15, 20], communication networks [11], and mobile robotics [1, 3, 8]. Among other things, it has been shown that given F Byzantine or malicious nodes, there exists a strategy for the misbehaving nodes to disrupt consensus if the network connectivity¹ is $2F$ or less. Conversely, if the network connectivity is at least $2F + 1$, then there exist strategies for the *normal* nodes to use that ensure consensus is reached [20, 23, 29]. However, these methods either require that normal nodes have at least some nonlocal information or assume that the network is *complete*, i.e., all-to-all communication or sensing [1, 3, 8, 15, 16]. Moreover, these algorithms tend to be computationally expensive. Therefore, there is a need for resilient consensus algorithms that are *low complexity* and *operate using only local information*.

Typically, an upper bound on the number of faults or threats in the network is assumed, i.e., at most F out of n nodes fail or are compromised. We refer to this *threat assumption*, or *scope of threat*, as the F -total model. In cases where it is preferable to make *no global assumptions*, we are interested in other threat assumptions that are strictly local. For example, whenever each node only assumes that at most F nodes in its *neighborhood* are compromised (but there is no other bound on the total number of compromised nodes), the scope of threat is F -local.

In addition to the *number* of misbehaving nodes, one can consider various *threat models* for the misbehaving nodes; examples include *non-colluding* [23], *malicious* [16, 23, 29], or *Byzantine* [1, 15, 17, 32] nodes. Non-colluding nodes are unaware of the network topology, which other nodes are misbehaving, or the states of non-neighbor nodes. On the other hand, malicious nodes have full

¹The network connectivity is defined as the smaller of the two following values: (i) the size of a minimal vertex cut and (ii) $n - 1$, where n is the number of nodes in the network.

knowledge of the networked system and therefore, worst case behavior must be assumed. The only difference between malicious and Byzantine nodes lies in their capacity for deceit. Malicious nodes are unable to convey different information to different neighbors in the network, whereas Byzantine nodes can.

Recently, we have studied resilient algorithms in the presence of misbehaving nodes. In [16], we introduce the Adversarial Robust Consensus Protocol (ARC-P) for consensus in the presence of malicious agents under the F -total model in continuous-time complete networks, with the agents also modeled in continuous time. The results of [16] are extended to both malicious and Byzantine threat models in networks with constrained information flow and dynamic network topology in [17]. In [34], we study general distributed algorithms with F -local malicious adversaries, encompassing ARC-P. In [17, 34], we show that traditional graph theoretic properties such as connectivity and degree, which have played a vital role in characterizing the resilience of distributed algorithms (see [20, 29]), are no longer adequate when the agents make purely local decisions (i.e., without knowing nonlocal aspects of the network topology). Instead, in [34] we introduce a novel topological property, referred to as *network robustness*, and show that this concept is highly effective at characterizing the ability of purely local algorithms to succeed. Separate sufficient and necessary conditions are provided in [34] for ARC-P to achieve resilient consensus in discrete time, and it is shown that the preferential attachment mechanism for generating complex networks produces robust graphs.

In this paper, we continue our study of resilient consensus in the presence of malicious nodes while using only local information. We are interested in partially constrained, asymptotic consensus in dynamic networks. To allow for multiple interpretations of the results, we formulate the problem in a setting common to discrete and continuous time for node dynamics and time-invariant or time-varying network topologies. We extend the Adversarial Robust Consensus Protocol (ARC-P) introduced in [16] to weighted networks. We then describe robust network topologies that are rich enough to enable resilience to malicious nodes, but are not too restrictive in terms of communication cost (i.e., number of communication links); in particular, we generalize the robustness property of [34]. Given these topological properties, we fully characterize the consensus behavior of the normal nodes using ARC-P under the F -total model of malicious nodes, and provide, for the first time, a necessary and sufficient condition for the algorithm to succeed. Additionally, for the F -local threat model, we provide improved separate necessary and sufficient conditions for asymptotic agreement of the normal nodes in the presence of malicious nodes.

The rest of the paper is organized as follows. Section 2 introduces the problem in a framework common to discrete and continuous time. Section 3 presents ARC-P in the unified framework. Section 4 motivates the need for robust network topologies and introduces the formal definitions. The main results are given in Section 5. A simulation example is presented in Section 6. Finally, some discussion is given in Section 7.

2. PROBLEM FORMULATION

Consider a time-varying network modeled by the (finite, simple) *directed graph*, or *digraph*, $\mathcal{D}[t] = \{\mathcal{V}, \mathcal{E}[t]\}$, where $\mathcal{V} = \{1, \dots, n\}$ is the *node set* and $\mathcal{E}[t] \subset \mathcal{V} \times \mathcal{V}$ is the *directed edge set* at time t . The node set is partitioned into a set of *normal nodes* \mathcal{N} and a set of *adversary nodes* \mathcal{M} . Note that $t \in \mathbb{R}_{\geq 0}$ for continuous time and $t \in \mathbb{Z}_{\geq 0}$ for discrete time. When we refer to both cases, we generically say *at time t* .

The time-varying topology of the network is governed by a piecewise constant switching signal $\sigma(\cdot)$, which is defined on $\mathbb{Z}_{\geq 0}$ for

discrete time and $\mathbb{R}_{\geq 0}$ for continuous time, and takes values in the set of all digraphs on n nodes. Let $\{\tau_k\}$, $k \in \mathbb{Z}_{>0}$ denote the set of switching instances. For continuous time, we assume that there exists some constant $\tau \in \mathbb{R}_{>0}$ such that $\tau_{k+1} - \tau_k \geq \tau$ for all $k \geq 0$. In other words, $\sigma(\cdot)$ is subject to the *dwell time* τ .

Each directed edge $(j, i) \in \mathcal{E}[t]$ models *information flow* and indicates that node i can be influenced by (or receive information from) node j at time t . The set of *in-neighbors*, or just *neighbors*, of node i at time t is defined as $\mathcal{V}_i[t] = \{j \in \mathcal{V} : (j, i) \in \mathcal{E}[t]\}$ and the (in-)degree of i is denoted $d_i^{\text{in}}[t] = |\mathcal{V}_i[t]|$. Likewise, the set of *out-neighbors* of node i at time t is defined as $\mathcal{V}_i^{\text{out}}[t] = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}[t]\}$. Because each node has access to its own state at time t , we also consider the *inclusive neighbors* of node i , denoted $\mathcal{I}_i[t] = \mathcal{V}_i[t] \cup \{i\}$. Note that time-invariant networks are represented simply by dropping the dependence on time t .

2.1 Update Model

Suppose that each node $i \in \mathcal{N}$ begins with some private value $x_i[0] \in \mathbb{R}$ (which could represent an opinion, vote, measurement, etc.). The nodes interact synchronously by conveying their value to (out-)neighbors in the network. Each normal node updates its own value over time according to a prescribed rule, which is modeled as

$$D[x_i[t]] = f_{i, \sigma(t)}(\{x_j[t]\}), \quad j \in \mathcal{I}_i[t], i \in \mathcal{N},$$

where $D[x_i[t]] = \dot{x}_i[t]$ is the *derivative operator* for continuous time and $D[x_i[t]] = x_i[t+1] - x_i[t]$ is the *forward difference operator* for discrete time. Each function $f_{i, \sigma(t)}(\cdot)$ can be arbitrary,² and may be different for each node, depending on its role in the network. These functions are designed *a priori* so that the normal nodes reach consensus. However, some of the nodes may not follow the prescribed strategy if they are compromised by an adversary. Such misbehaving nodes threaten the group objective, and it is important to design the $f_{i, \sigma(t)}(\cdot)$'s in such a way that the influence of such nodes can be eliminated or reduced without prior knowledge about their identities.

2.2 Threat Model

DEFINITION 1. A node $k \in \mathcal{M}$ is said to be **malicious** if

- it is not normal (i.e., it does not follow the prescribed update model either for at least one time-step in discrete time, or for some time interval of nonzero Lebesgue measure in continuous time);
- it conveys the same value, $x_k[t]$, to each out-neighbor;
- (for continuous-time systems) its value trajectory, $x_k[t] \forall t$, is a uniformly continuous function of time.

A few remarks are in order concerning malicious nodes. First, each malicious node is allowed to be omniscient (i.e., it knows all other values and the full network topology; it is aware of the update rules $f_{i, \sigma(t)}(\cdot)$, $\forall i \in \mathcal{N}$; it knows which other nodes are adversaries; and it knows the plans of the other adversaries). The statement in the definition that the malicious nodes are not normal is intended to capture the idea that they do not apply the prescribed update rule for all time. The second assumption is intended as an assertion on the network realization. That is, if the network is realized through sensing or broadcast communication, it is assumed that the out-neighbors receive the same information. The third point is a technical assumption that applies only to malicious

²In continuous time, $f_{i, \sigma(t)}(\cdot)$ must satisfy appropriate assumptions to ensure existence of solutions.

nodes modeled in continuous time. Limited only by these assumptions, the malicious nodes are otherwise allowed to operate in an arbitrary (potentially worst case) manner.

2.3 Scope of Threats

To be more precise, we formally define the scope of the threats. While there are various stochastic models that could be used to formalize the threat assumptions, here we use a deterministic approach and consider upper bounds on the number of compromised nodes either in the network (F -total) or in each node's neighborhood (F -local).

DEFINITION 2 (F -TOTAL SET). A set $\mathcal{S} \subset \mathcal{V}$ is **F -total** if it contains at most F nodes in the network, i.e., $|\mathcal{S}| \leq F$, $F \in \mathbb{Z}_{\geq 0}$.

DEFINITION 3 (F -LOCAL SET). A set $\mathcal{S} \subset \mathcal{V}$ is **F -local** if it contains at most F nodes in the neighborhood of the other nodes for all t , i.e., $|\mathcal{V}_i[t] \cap \mathcal{S}| \leq F$, $\forall i \in \mathcal{V} \setminus \mathcal{S}$, $F \in \mathbb{Z}_{\geq 0}$.

It should be noted that because the network topology may be time-varying, the local properties defining an F -local set must hold at all time instances. These definitions facilitate the definitions of the scope of threat models.

DEFINITION 4. A set of adversary nodes is **F -totally bounded** or **F -locally bounded** if it is an F -total set or F -local set, respectively. We refer to these threat scopes as the **F -total** and **F -local** models, respectively.

F -totally bounded faults have been studied in distributed computing [15, 20, 32] and mobile robotics [1, 3, 8] for both stopping (or crash) failures and Byzantine failures. The F -locally bounded fault model has been studied in the context of fault-tolerant broadcasting [12, 24].

2.4 Resilient Asymptotic Consensus

Given the threat model and scope of threats, we formally define resilient asymptotic consensus. Let $M_{\mathcal{N}}[t]$ and $m_{\mathcal{N}}[t]$ be the *maximum* and *minimum* values of the normal nodes at time t , respectively.

DEFINITION 5 (RESILIENT ASYMPTOTIC CONSENSUS). The normal nodes are said to achieve **resilient asymptotic consensus** in the presence of (a) F -totally bounded, or (b) F -locally bounded misbehaving nodes if $|x_i[t] - x_j[t]| \rightarrow 0$ as $t \rightarrow \infty$, for all $i, j \in \mathcal{N}$, the normal values converge to a point (i.e., the consensus value) for any choice of initial values, and the normal values remain in the interval $[m_{\mathcal{N}}[0], M_{\mathcal{N}}[0]]$ for all t . Whenever the scope of threat is understood, we simply say that the normal nodes reach **asymptotic consensus**.

The resilient asymptotic consensus problem has two important conditions. First, the normal nodes must reach asymptotic consensus in the presence of misbehaving nodes given a particular threat model (e.g., malicious node) and scope of threat (e.g., F -total). This is a condition on agreement. Additionally, it is required that the interval containing the initial values of the normal nodes is an invariant set for the normal nodes, and that the consensus value to which the consensus process converges lies within this interval. This condition is a validity or safety condition that constrains the consensus quantity.

3. CONSENSUS ALGORITHM

Linear consensus algorithms have attracted significant interest in recent years [22, 26], due to their applicability in a variety of

contexts. In such strategies, at time t , each node senses or receives information from its neighbors, and changes its value according to

$$D[x_i[t]] = \sum_{j \in \mathcal{J}_i[t]} w_{ij}[t] x_j[t], \quad (1)$$

where $w_{ij}[t]$ is the weight assigned to node j 's value by node i at time t .

Different conditions have been reported in the literature to ensure asymptotic consensus is reached [13, 21, 25, 31, 33]. In discrete time, it is common to assume that there exists a constant $\alpha \in \mathbb{R}$, $0 < \alpha < 1$ such that all of the following conditions hold:³

- $w_{ij}[t] = 0$ whenever $j \notin \mathcal{J}_i[t]$, $i \in \mathcal{N}$, $t \in \mathbb{Z}_{\geq 0}$;
- $w_{ij}[t] \geq \alpha$, $\forall j \in \mathcal{V}_i[t]$, $i \in \mathcal{N}$, $t \in \mathbb{Z}_{\geq 0}$;
- $w_{ii}[t] \geq \alpha - 1$, $\forall i \in \mathcal{N}$, $t \in \mathbb{Z}_{\geq 0}$;
- $\sum_{j=1}^n w_{ij}[t] = 0$, $\forall i \in \mathcal{N}$, $t \in \mathbb{Z}_{\geq 0}$.

In continuous time there are similar conditions, except in this case the self-weights are given by

$$w_{ii}[t] = - \sum_{j \in \mathcal{V}_i[t]} w_{ij}[t], \quad \forall i \in \mathcal{N}, \forall t \in \mathbb{R}_{\geq 0}.$$

In this case, to make sure the weights are uniformly bounded, we additionally assume $\exists \beta \in \mathbb{R}_{>0}$, $\beta \geq \alpha$, such that $w_{ij}[t] \leq \beta$, for all $i, j \in \mathcal{N}$ and $t \in \mathbb{R}_{\geq 0}$. Similar to the discrete time case, the weights $w_{ij}[t]$ are zero precisely whenever $j \notin \mathcal{J}_i[t]$, and bounded below by α otherwise. Together, these conditions imply the analogue of the fourth condition above.

Given these conditions, a necessary and sufficient condition for reaching asymptotic consensus in time-invariant networks is that the digraph has a *rooted out-branching*, also called a *rooted directed spanning tree* [26]. The case of dynamic networks is not quite as straightforward. In this case, under the conditions stated above, a sufficient condition for reaching asymptotic consensus is that there exists a uniformly bounded sequence of contiguous time intervals such that the union of digraphs across each interval has a rooted out-branching [25]. Recently, a more general condition referred to as the *infinite flow property* has been shown to be both necessary and sufficient for asymptotic consensus for a class of discrete-time stochastic models [30]. Finally, the lower bound on the weights is needed because there are examples of asymptotically vanishing weights in which consensus is not reached [19].

In general, the problem of selecting the best weights in the linear update rule (1) is nontrivial, and the choice affects the rate of consensus. The problem of selecting the optimal weights (with respect to the speed of the consensus process) in time-invariant, discrete-time, bidirectional networks is addressed in [33] by formulating a semidefinite program (SDP). However, this SDP is solved at design time with global knowledge of the network topology. A simple choice of weights for discrete time systems that requires only local information is to let $w_{ij}[t] = 1/(1 + d_i^{\text{in}}[t])$ for $j \in \mathcal{V}_i[t]$ and $w_{ii}[t] = -d_i^{\text{in}}[t]/(1 + d_i^{\text{in}}[t])$. In continuous time, a simple choice is to let $w_{ij} \equiv 1$ for $j \in \mathcal{V}_i[t]$ and $w_{ii}[t] = -d_i^{\text{in}}[t]$.

One problem with the linear update given in (1) is that it is not resilient to misbehaving nodes. In fact, it was shown in [10, 13] that a single 'leader' node can cause all agents to reach consensus on an

³The conditions on the weights are modified from what is reported in the literature to account for the forward difference operator. Accounting for this, the updated value of each node is formed as a convex combination of the neighboring values and its own value.

arbitrary value of its choosing (potentially resulting in a dangerous situation in physical systems).

The Adversarial Robust Consensus Protocol (ARC-P) addresses this vulnerability of the linear update of (1) by a simple modification. Instead of trusting every neighbor by using every value in the update, the normal node first removes the extreme values from consideration in the update by effectively setting their weights (temporarily) to zero. It is shown in subsequent sections that this simple strategy provides resilience against malicious nodes in robust networks.

3.1 Description of ARC-P

At time t , each normal node i obtains the values of other nodes in its neighborhood. At most F of node i 's neighbors may be malicious; however, node i is unsure of which neighbors may be compromised. To ensure that node i updates its value in a safe manner, it removes the extreme values with respect to its own value according to the following protocol.

1. At time t , each normal node i obtains the values of its neighbors, and forms a sorted list.
2. If there are less than F values strictly larger than its own value, $x_i[t]$, then normal node i removes all values that are strictly larger than its own. Otherwise, it removes precisely the largest F values in the sorted list (breaking ties arbitrarily). Likewise, if there are less than F values strictly smaller than its own value, then node i removes all values that are strictly smaller than its own. Otherwise, it removes precisely the smallest F values.
3. Let $\mathcal{R}_i[t]$ denote the set of nodes whose values were removed by normal node i in step 2 at time t . Each normal node i applies the update

$$D[x_i[t]] = \sum_{j \in \mathcal{J}_i[t] \setminus \mathcal{R}_i[t]} w_{ij}[t] x_j[t], \quad (2)$$

where the weights $w_{ij}[t]$ satisfy the conditions stated above, but with $\mathcal{J}_i[t]$ replaced by $\mathcal{J}_i[t] \setminus \mathcal{R}_i[t]$.⁴

As a matter of terminology, we refer to the largest number of values that a node could throw away as the *parameter* of the algorithm. Above, the parameter of ARC-P with the F -local and F -total models is $2F$.

Observe that the set of nodes removed by normal node i , $\mathcal{R}_i[t]$, is possibly time-varying. Hence, even though the underlying network topology may be fixed, ARC-P effectively induces switching behavior, and can be viewed as the linear update of (1) with a specific rule for state-dependent switching (the rule given in step 2).

4. ROBUST NETWORK TOPOLOGIES

4.1 Network Robustness

In this section, we introduce *robust network topologies* that satisfy certain graph theoretic properties, which we refer to generically as *network robustness*. Network robustness formalizes the notion of sufficient redundancy of information flow to subsets of a network in a single hop. Therefore, this property holds promise to be effective for the study of resilient distributed algorithms that use

⁴In this case, a simple choice for the weights in discrete time is to let $w_{ij}[t] = 1/(1 + d_i^{\text{in}}[t] - |\mathcal{R}_i[t]|)$ for $j \in \mathcal{V}_i[t]$ and $w_{ii}[t] = (|\mathcal{R}_i[t]| - d_i^{\text{in}}[t])/(1 + d_i^{\text{in}}[t] - |\mathcal{R}_i[t]|)$. In continuous time, let $w_{ij} \equiv 1$ for $j \in \mathcal{V}_i[t]$ and $w_{ii}[t] = |\mathcal{R}_i[t]| - d_i^{\text{in}}[t]$.

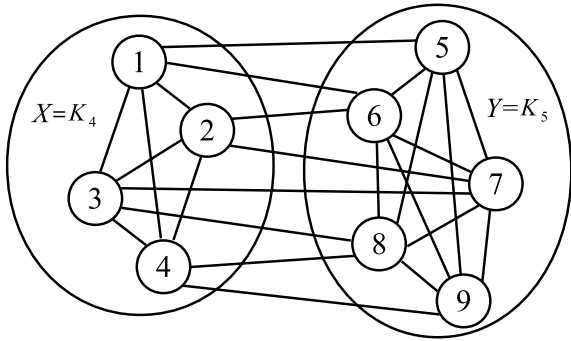


Figure 1: Example of a 5-connected graph satisfying Prop. 1 whenever $F = 2$.

only local information. In contrast, network connectivity formalizes the notion of sufficient redundancy of information flow across the network through independent paths. Due to the fact that each independent path may include multiple intermediate nodes, network connectivity is well-suited for studying resilient distributed algorithms that assume such nonlocal information is available (for example, by explicitly relaying information across multiple hops in the network [20], or by ‘inverting’ the dynamics on the network to recover the needed information [23, 29]). However, network connectivity is no longer an appropriate metric for an algorithm that uses purely local information, such as ARC-P. This is demonstrated by the following proposition [34].

PROPOSITION 1. *There exists a graph with connectivity $\kappa = \lfloor \frac{n}{2} \rfloor + F - 1$ in which ARC-P does not ensure asymptotic consensus.*

Figure 1 illustrates an example of this kind of graph with $n = 9$, $F = 2$, and $\kappa = 5$. In this graph, there are two cliques (complete subgraphs), $X = K_4$ and $Y = K_5$, where K_n is the complete graph on n nodes. Each node in X has exactly $F = 2$ neighbors in Y , and all but two nodes in Y have $F = 2$ neighbors in X (nodes 5 and 9 have only one neighbor in X , because otherwise a node in X would have more than $F = 2$ neighbors in Y). One can see that if the initial values of nodes in X and Y are $a \in \mathbb{R}$ and $b \in \mathbb{R}$, respectively, with $a \neq b$, then asymptotic consensus is not achieved whenever ARC-P is used with parameter $2F$ – even in the absence of misbehaving nodes. This is because each node views the values of its F neighbors from the opposing set as extreme, and removes all of these values from its list. The only remaining values for each node are from its own set, and thus no node ever changes its value.

The situation can be even worse in the more general case of digraphs. Examples of digraphs are illustrated in [17] that are $(n-1)$ -connected and have minimum out-degree $n - 2$, yet ARC-P still cannot guarantee asymptotic consensus. Thus, even digraphs with a relatively large connectivity (or minimum out-degree) are not sufficient to guarantee consensus of the normal nodes, indicating the inadequacy of these traditional metrics to analyze the convergence properties of ARC-P. Taking a closer look at the graph in Fig. 1, we see that the reason for the failure of consensus is that no node has enough neighbors in the opposite set; this causes every node to throw away all useful information from outside of its set, and prevents consensus. Based on this intuition, the following properties, i.e., r -reachable sets and r -robustness, were introduced in [34].

DEFINITION 6 (r -REACHABLE SET). *Given a digraph \mathcal{D} and a subset \mathcal{S} of nodes of \mathcal{D} , we say \mathcal{S} is an r -reachable set if $\exists i \in \mathcal{S}$ such that $|\mathcal{V}_i \setminus \mathcal{S}| \geq r$, where $r \in \mathbb{Z}_{\geq 0}$.*

A set \mathcal{S} is r -reachable if it contains a node that has at least r neighbors outside of \mathcal{S} . The parameter r quantifies the redundancy of information flow from nodes outside of \mathcal{S} to some node inside \mathcal{S} . Intuitively, the r -reachability property captures the idea that some node inside the set is influenced by a sufficiently large number of nodes from outside the set. The above reachability property pertains to a given set \mathcal{S} ; in order to generalize this notion of redundancy to the entire network, we introduce the following definition of r -robustness.

DEFINITION 7 (r -ROBUSTNESS). *A digraph $\mathcal{D} = \{\mathcal{V}, \mathcal{E}\}$ is r -robust if for every pair of nonempty, disjoint subsets of \mathcal{V} , at least one of the subsets is r -reachable.*

The reason that pairs of nonempty, disjoint subsets of nodes are considered in the definition of r -robustness can be seen in the example of Fig. 1. If either X or Y were 3-reachable ($r = F + 1 = 3$), then at least one node would be sufficiently influenced by a node outside of its set in order to drive it away from the values of its group, and thereby lead its group to the values of the other set. However, if there are misbehaving nodes in the network, then the situation becomes more complex. For example, consider the F -total model of malicious nodes, and consider two sets X and Y in the graph. Let s be the total number of nodes in these two sets that each have at least $F + 1$ neighbors outside their own set. If $s \leq F$, then simply by choosing these nodes to be malicious, the sets X and Y contain no normal nodes that bring in enough information from outside, and thus the system can be prevented from reaching consensus. This reasoning suggests a need to specify a minimum number of nodes that are sufficiently influenced from outside of their set (in this example, at least $F + 1$ nodes). This intuition leads to the following generalizations of r -reachability and r -robustness.

DEFINITION 8 ((r, s) -REACHABLE SET). *Given a digraph \mathcal{D} and a subset of nodes \mathcal{S} , we say that \mathcal{S} is an (r, s) -reachable set if there are at least s nodes in \mathcal{S} with at least r neighbors outside of \mathcal{S} , where $r, s \in \mathbb{Z}_{\geq 0}$; i.e., given $\mathcal{X}_{\mathcal{S}} = \{i \in \mathcal{S} : |\mathcal{V}_i \setminus \mathcal{S}| \geq r\}$, then $|\mathcal{X}_{\mathcal{S}}| \geq s$.*

Observe that r -reachability is equivalent to $(r, 1)$ -reachability; hence, (r, s) -reachability is a strict generalization of r -reachability. If a set \mathcal{S} is (r, s) -reachable, we know there are at least s nodes in \mathcal{S} with at least r neighbors outside of \mathcal{S} . Thus, if \mathcal{S} is (r, s) -reachable, then it is (r, s') -reachable, for $s' \leq s$. Also, it is clear that $s \leq |\mathcal{S}|$ and all subsets of nodes of any digraph are $(r, 0)$ -reachable. The additional specificity on the number of nodes with redundant information flow from outside of their set is useful for defining a more general notion of robustness.

DEFINITION 9 ((r, s) -ROBUSTNESS). *A digraph $\mathcal{D} = \{\mathcal{V}, \mathcal{E}\}$ is (r, s) -robust if for every pair of nonempty, disjoint subsets \mathcal{S}_1 and \mathcal{S}_2 of \mathcal{V} such that \mathcal{S}_1 is $(r, s_{r,1})$ -reachable and \mathcal{S}_2 is $(r, s_{r,2})$ -reachable with $s_{r,1}$ and $s_{r,2}$ maximal (i.e., $s_{r,k} = |\mathcal{X}_{\mathcal{S}_k}|$ where $\mathcal{X}_{\mathcal{S}_k} = \{i \in \mathcal{S}_k : |\mathcal{V}_i \setminus \mathcal{S}_k| \geq r\}$ for $k \in \{1, 2\}$), then at least one of the following hold:*

- (i) $s_{r,1} = |\mathcal{S}_1|$;
- (ii) $s_{r,2} = |\mathcal{S}_2|$;
- (iii) $s_{r,1} + s_{r,2} \geq s$.

A few remarks are in order with respect to this definition. The (r, s) -robustness property generally aims to capture the idea that “enough” nodes in the sets \mathcal{S}_1 and \mathcal{S}_2 have at least r neighbors outside of their respective sets, for all nonempty and disjoint $\mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{V}$.

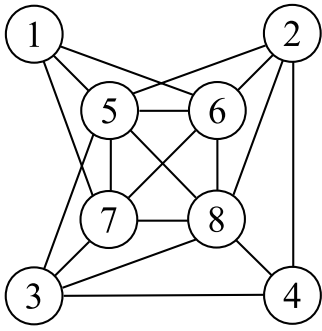


Figure 2: A 3-robust graph that is not (3,2)-robust.

\mathcal{V} . In order to specify what is meant by “enough” nodes, it is necessary to take the maximal $s_{r,k}$ for which \mathcal{S}_k is $(r, s_{r,k})$ -reachable with $k \in \{1, 2\}$ (since \mathcal{S}_k is $(r, s'_{r,k})$ -reachable for $s'_{r,k} \leq s_{r,k}$). Clearly, if $s_{r,k} = |\mathcal{S}_k|$ for either $k \in \{1, 2\}$, then *all* nodes in \mathcal{S}_k have at least r neighbors outside of \mathcal{S}_k , in which case at least one of conditions (i) or (ii) is satisfied, and we say there are “enough” nodes. Alternatively, if there are at least s nodes with at least r neighbors outside of their respective sets in the union $\mathcal{S}_1 \cup \mathcal{S}_2$, then condition (iii) is satisfied, and we say there are “enough” nodes. The reason to have multiple interpretations of what constitutes “enough” nodes is to be able to state the property uniformly over all nonempty and disjoint pairs of subsets of nodes. Clearly, if $|\mathcal{S}_1 \cup \mathcal{S}_2| < s$, then condition (iii) cannot be satisfied. More generally, in many cases where $\min\{|\mathcal{S}_1|, |\mathcal{S}_2|\} < s$, it is also not possible to satisfy condition (iii); e.g., whenever $\mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{V}$ and $r \geq s$. On the other hand, for relatively large sets (i.e., $\min\{|\mathcal{S}_1|, |\mathcal{S}_2|\} \geq s$), conditions (i) and (ii) do imply condition (iii).

An important observation is that $(r, 1)$ -robustness is equivalent to r -robustness. This holds because conditions (i)–(iii) for $(r, 1)$ -robustness collapse to the condition that at least one of \mathcal{S}_1 and \mathcal{S}_2 is r -reachable. In general, a digraph is (r, s') -robust if it is (r, s) -robust for $s' \leq s$; therefore, a digraph is r -robust whenever it is (r, s) -robust, for $s \geq 1$. The converse, however, is not true. Consider the graph in Fig. 2. This graph is 3-robust, but is not (3, 2)-robust. For example, let $\mathcal{S}_1 = \{1, 3, 5, 6, 7\}$ and $\mathcal{S}_2 = \{2, 4\}$. Thus, only node 2 has at least 3 nodes outside of its set, so all of the conditions (i)–(iii) fail. Therefore, (r, s) -robustness is a strict generalization of r -robustness.

Next, consider again the example of Fig. 1. It can be shown that this graph is $(2, s)$ -robust, for all $s \in \mathbb{Z}_{\geq 0}$. This follows because *all* nodes in at least one of the sets \mathcal{S}_1 and \mathcal{S}_2 have at least 2 neighbors outside of their set, for any nonempty and disjoint $\mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{V}$. Therefore, condition (iii) in the definition *never* needs to hold true, and the definition is satisfied with $r = 2$ for all $s \in \mathbb{Z}_{\geq 0}$. It is rather atypical, in general, for digraphs to satisfy (r, s) -robustness for all $s \in \mathbb{Z}_{\geq 0}$; however, it can be the case for graphs with high connectivity and small diameter.

On the other hand, the graph in Fig. 1 is *not* 3-robust. This can be shown by selecting $\mathcal{S}_1 = X$ and $\mathcal{S}_2 = Y$. Note that an (r, s) -robust digraph is (r', s) -robust for $r' \leq r$. The question then arises, how does one compare relative robustness between digraphs? Clearly, if digraph \mathcal{D}_1 is (r_1, s_1) -robust and digraph \mathcal{D}_2 is (r_2, s_2) -robust with maximal r_k and s_k ⁵ for $k \in \{1, 2\}$, where $r_1 > r_2$ and $s_1 > s_2$, then one can conclude that \mathcal{D}_1 is more

⁵We adopt the convention that given a digraph that is (r, s) -robust for all $s \in \mathbb{Z}_{\geq 0}$, its maximal s is ∞ .

robust than \mathcal{D}_2 . However, in cases where $r_1 > r_2$ but $s_1 < s_2$, which digraph is more robust? For example, the graph of Fig. 1 is $(2, s)$ -robust for all $s \in \mathbb{Z}_{\geq 0}$, but is not 3-robust, whereas the graph in Fig. 2 is 3-robust, but is not (2,5)-robust (e.g., let $\mathcal{S}_1 = \{1, 5, 6\}$ and $\mathcal{S}_2 = \{2, 3, 4\}$). In general, the r -robustness property takes precedence in the partial order that determines relative robustness, and the maximal s in (r, s) -robustness is used for finer grain partial ordering (i.e., ordering the robustness of two r -robust digraphs with the same value of r). Therefore, the graph in Fig. 2 is more robust than the graph of Fig. 1. Yet, the graph of Fig. 2 is only 3-connected, whereas the graph of Fig. 1 is 5-connected. Hence, it is possible that a digraph with *less* connectivity is *more* robust.

We demonstrate in Section 5 that the r -robustness property is useful for analyzing ARC-P with parameter $2F$ under the F -local model, and show that (r, s) -robustness is the key property for analyzing ARC-P with parameter $2F$ under the F -total model. More specifically, we show that $(F+1, F+1)$ -robustness of the network is both necessary and sufficient for normal nodes using ARC-P with parameter $2F$ to achieve resilient asymptotic consensus whenever the scope of threat is F -total, the threat model is malicious, and the network is time-invariant. Likewise, we show that $(2F+1)$ -robustness of the network is sufficient for ARC-P with parameter $2F$ to achieve resilient asymptotic consensus whenever the scope of threat is F -local.

4.2 Construction of Robust Digraphs

Note that robustness requires checking every possible nonempty disjoint pair of subsets of nodes in the digraph for certain conditions. Currently, we do not have a computationally efficient method to check whether these properties hold in arbitrary digraphs. However, in [34] it is shown that the common *preferential-attachment* model for complex networks (e.g., [2]) produces r -robust graphs, provided that a sufficient number of links are added to the network as new nodes are attached. In this subsection, we extend this construction to show that preferential attachment also leads to (r, s) -robust graphs.

THEOREM 1. *Let $\mathcal{D} = \{\mathcal{V}, \mathcal{E}\}$ be an (r, s) -robust digraph (with $s \geq 1$). Then the digraph $\mathcal{D}' = \{\mathcal{V} \cup \{v_{new}\}, \mathcal{E} \cup \mathcal{E}_{new}\}$, where v_{new} is a new vertex added to \mathcal{D} and \mathcal{E}_{new} is the directed edge set related to v_{new} , is (r, s) -robust if $d_{v_{new}}^{in} \geq r + s - 1$.*

The above result indicates that to construct an (r, s) -robust digraph with n nodes (where $n > r$), we can start with an (r, s) -robust digraph with relatively smaller order (such as a complete graph), and continually add new nodes with incoming edges from at least $r + s - 1$ nodes in the existing digraph. Note that this method does not specify *which* existing nodes should be chosen. The preferential-attachment model corresponds to the case when the nodes are selected with a probability proportional to the number of edges that they already have. This leads to the formation of so-called *scale-free* networks [2], and is cited as a plausible mechanism for the formation of many real-world complex networks. Theorem 1 indicates that a large class of scale-free networks are resilient to the threat models studied in this paper (provided the number of edges added in each round is sufficiently large when the network is forming).

For example, Fig. 3 illustrates a (3, 2)-robust graph constructed using the preferential attachment model starting with the complete graph on 5 nodes, K_5 (which is also (3,3)-robust and is the only (3,2)-robust digraph on 5 nodes), and with 4 new edges added to each new node. Note that this graph is also 4-robust, which could *not* be predicted from Theorem 1 since K_5 is not 4-robust. Therefore, it is actually possible (but not guaranteed) to end up with

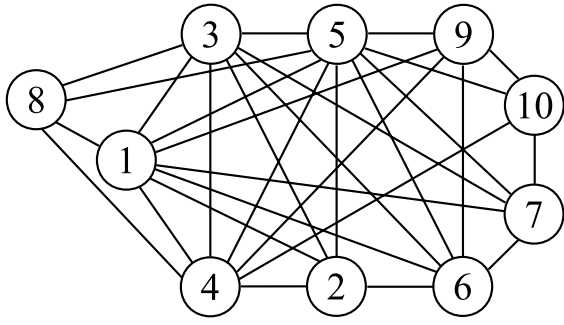


Figure 3: A (3,2)-robust graph constructed from K_5 using preferential attachment.

a more robust digraph than the initial one using the preferential-attachment growth model.

5. RESILIENT CONSENSUS RESULTS

In this section, we provide the key results showing that sufficiently robust digraphs guarantee resilient consensus. Due to space limitations, the proofs are omitted here; instead, we provide a brief outline of the arguments. The full proofs are given in [18].

5.1 F -Total Model

THEOREM 2. *Consider a time-invariant network modeled by a directed graph $\mathcal{D} = \{\mathcal{V}, \mathcal{E}\}$. In the presence of malicious nodes under the F -total model, ARC-P with parameter $2F$ achieves resilient asymptotic consensus if and only if the network topology is $(F + 1, F + 1)$ -robust.*

Outline of proof:

(Sufficiency) Recall that $M_{\mathcal{N}}[t]$ and $m_{\mathcal{N}}[t]$ are the maximum and minimum values of the normal nodes at time t , respectively. Define $\Phi[t] = M_{\mathcal{N}}[t] - m_{\mathcal{N}}[t]$ and note that $\Phi[t] \rightarrow 0$ as $t \rightarrow \infty$ if and only if the normal nodes reach asymptotic consensus. Since Φ is a non-increasing function of t , the main idea of the proof is to show that after some bounded time t_c , $\Phi[t]$ will shrink by a certain nontrivial fraction $0 < c_t < 1$, i.e. $\Phi[t + t_c] \leq c_t \Phi[t], \forall t$. To show this, the $(F + 1, F + 1)$ -robustness property is used to show that there exists a chain of subsets of nodes in either the subset of normal nodes with maximum value, or the subset of normal nodes with minimum value such that the nodes in the first subset in the chain have enough neighbors with values smaller (or larger) than their own in order to drive their values away from the extreme value. Then, all nodes in the next subset in the chain are guaranteed to have enough neighbors to drive their values away from the extreme value, and so on, until the last of the extreme values are shifted by a nontrivial amount. The uniform continuity of the malicious nodes' value trajectories and the bounds on the weights are used in order to formalize the argument.

(Necessity) If \mathcal{D} is not $(F + 1, F + 1)$ -robust, then there are nonempty, disjoint $\mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{V}$ such that none of the conditions (i) – (iii) hold. Suppose the initial value of each node in \mathcal{S}_1 is a and each node in \mathcal{S}_2 is b , with $a < b$. Let all other nodes have initial values taken from the interval (a, b) . Since $s_{F+1,1} + s_{F+1,2} \leq F$, suppose all nodes in $\mathcal{X}_{\mathcal{S}_1}$ and $\mathcal{X}_{\mathcal{S}_2}$ are malicious and keep their values constant. With this assignment of adversaries, there is still at least one normal node in both \mathcal{S}_1 and \mathcal{S}_2 since $s_{F+1,1} < |\mathcal{S}_1|$ and $s_{F+1,2} < |\mathcal{S}_2|$, respectively. Since these normal nodes remove the F or less values of in-neighbors outside of their respective sets, no consensus among normal nodes is reached. \square

When the network is time-varying, one can state the following corollary of the above theorem.

COROLLARY 1. *Consider a time-varying network modeled by a directed graph $\mathcal{D}[t] = \{\mathcal{V}, \mathcal{E}[t]\}$. In the presence of malicious nodes under the F -total model, ARC-P with parameter $2F$ achieves resilient asymptotic consensus if there exists $t_0 \geq 0$ such that $\mathcal{D}[t]$ is $(F + 1, F + 1)$ -robust, $\forall t \geq t_0$.*

Outline of proof: The proof is similar to the proof of sufficiency of Theorem 2. In continuous time, the dwell time assumption is used by constructing the bounded time horizon t_c over which we are guaranteed that $\Phi[t]$ will shrink by a nontrivial fraction so that $t_c < \tau/2$. But, at any time t , the digraph may switch before time $t + t_c$, which may disrupt the chain argument made in the time-invariant case. But, since $t_c < \tau/2$, we may reconstruct a new chain at the switching instance, and are then assured that by time $t + 2t_c$, $\Phi[t]$ will shrink by a certain nontrivial fraction. \square

To illustrate these results on the examples of Section 4, the graphs in Figs. 1, 2, and 3 can withstand the compromise of at most 1 malicious node in the network using ARC-P with parameter $2F = 2$ (each graph is (2,2)-robust but not (3,3)-robust). This is not to say that it is impossible for the normal nodes to reach consensus if there are, for example, two nodes that are compromised. Instead, these results say that it is not possible that any two nodes can be compromised and still guarantee resilient asymptotic consensus using ARC-P with parameter $2F = 4$.

5.2 F -Local Model

THEOREM 3. *Consider a time-invariant network modeled by a directed graph $\mathcal{D} = \{\mathcal{V}, \mathcal{E}\}$. In the presence of malicious nodes under the F -local model, ARC-P with parameter $2F$ achieves resilient asymptotic consensus if the topology of the network is $(2F + 1)$ -robust; furthermore, a necessary condition is for the topology of the network to be $(F + 1)$ -robust.*

Outline of proof: The sufficiency proof in this case is similar to the proof of Theorem 2. The necessity proof is given in [34]. \square

COROLLARY 2. *Consider a time-varying network modeled by a directed graph $\mathcal{D}[t] = \{\mathcal{V}, \mathcal{E}[t]\}$. In the presence of malicious nodes under the F -local model, ARC-P with parameter $2F$ achieves resilient asymptotic consensus if there exists $t_0 \geq 0$ such that the topology of the network $\mathcal{D}[t]$ is $(2F + 1)$ -robust, $\forall t \geq t_0$.*

To illustrate these results, consider the 3-robust graph of Fig. 2. Recall that this graph cannot generally sustain 2 malicious nodes as specified by the 2-total model; it is not (3,3)-robust. However, under the 1-local model, it can sustain two malicious nodes if the right nodes are compromised. For example, nodes 1 and 4 may be compromised under the 1-local model and the normal nodes will still reach consensus. This example illustrates the advantage of the F -local model, where there is no concern about global assumptions. If a digraph is $(2F + 1)$ -robust, then up to F nodes may be compromised in any node's neighborhood, possibly resulting in more than F malicious nodes in the network (as in the previous example).

6. SIMULATIONS

This section presents a numerical example to illustrate our results. In this example, the network is given by the (2,2)-robust

graph shown in Fig. 4. Since the network is (2,2)-robust, it can sustain a single malicious node in the network under the 1-total model. Suppose that the node with the largest degree, node 14, is compromised and turns malicious. The nodes have continuous dynamics and the normal nodes use either the Linear Consensus Protocol (LCP) given in (1) or ARC-P for their control input. In either case, the weights are selected to be unity for all neighboring nodes that are kept, with the self-weights selected as $-d_i^{\text{in}}$ for LCP and $|\mathcal{R}_i[t]| - d_i^{\text{in}}$ for ARC-P for each normal node $i \in \mathcal{N}$. The initial values of the nodes are shown in Fig. 4 beneath the label of the node's value. The goal of the malicious agent is to drive the values of the normal nodes to a value of 2.

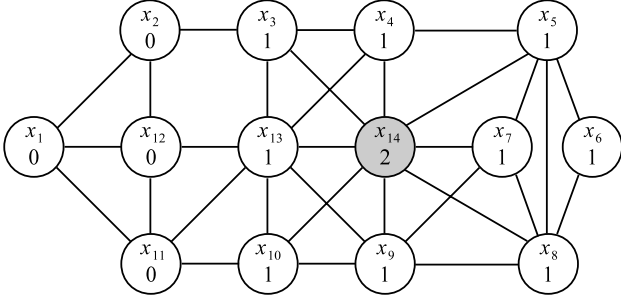
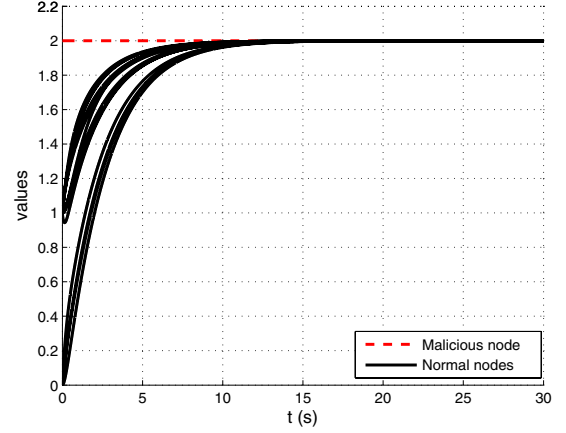


Figure 4: (2,2)-Robust Network topology.

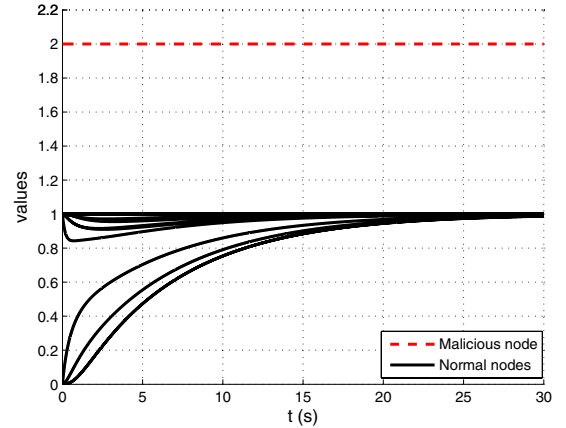
The results for this example are shown in Fig. 5. It is clear in Fig. 5(a) that the malicious node is able to drive the values of the normal nodes to its value of 2 whenever LCP is used. On the other hand, the malicious node is unable to achieve its goal whenever ARC-P is used. Note that due to the large degree of the malicious node, it has the potential to drive the consensus process to any value in the interval $[0, 1]$ by choosing the desired value as its initial value and remaining constant. However, this is allowed with resilient asymptotic consensus (because the consensus value is within the range of the initial values held by normal nodes). Another observation is that the consensus process in the case of ARC-P is slower than LCP; this is to be expected, due to the fact that ARC-P effectively removes certain edges from the network at each time-step. Finally, we remark that the chain argument sketched in the outline of the proof of Theorem 2 is demonstrated in Fig. 5(b). To see this, denote the set of normal nodes with initial value 0 as \mathcal{S}_0 and the set of normal nodes with initial value 1 as \mathcal{S}_1 . Then, $\mathcal{S}_0 = \{1, 2, 11, 12\}$ and $\mathcal{S}_1 = \{3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13\}$. Construct the following chain of subsets of \mathcal{S}_0 : $\mathcal{T}_0^1 = \{11\}$, $\mathcal{T}_0^2 = \{12\}$, $\mathcal{T}_0^3 = \{1, 2\}$. These subsets of \mathcal{S}_0 are defined recursively by the following steps:

1. Let \mathcal{T}_0^1 include all nodes in \mathcal{S}_0 that have at least $F + 1 = 2$ neighbors outside of \mathcal{S}_0 (in this case, node 11).
2. Form $\mathcal{S}_0^1 = \mathcal{S}_0 \setminus \mathcal{T}_0^1$.
3. Let \mathcal{T}_0^2 include all nodes in \mathcal{S}_0^1 that have at least $F + 1 = 2$ neighbors outside of \mathcal{S}_0^1 (in this case, node 12).
4. Form $\mathcal{S}_0^2 = \mathcal{S}_0^1 \setminus \mathcal{T}_0^2$.
5. Let \mathcal{T}_0^3 include all nodes in \mathcal{S}_0^2 that have at least $F + 1 = 2$ neighbors outside of \mathcal{S}_0^2 (in this case, nodes 1 and 2).
6. Form $\mathcal{S}_0^3 = \mathcal{S}_0^2 \setminus \mathcal{T}_0^3$.
7. Quit whenever $\mathcal{S}_0^k = \emptyset$ or $\mathcal{T}_0^k = \emptyset$ (in this case $\mathcal{S}_0^3 = \emptyset$ and $k = 3$).

In a similar manner, construct the following chain of subsets of \mathcal{S}_1 : $\mathcal{T}_1^1 = \{13\}$, $\mathcal{T}_1^2 = \{3, 10\}$, $\mathcal{T}_1^3 = \{4, 9\}$, $\mathcal{T}_1^4 = \emptyset$. Unlike the previous chain, this chain terminates with $\mathcal{S}_1^k \neq \emptyset$. For this reason, not all normal values in \mathcal{S}_1 are driven from their initial value of 1. Now, with these constructions, we can see the trajectories of the values of nodes in each subset in Fig. 5(b). For example, node 11 in \mathcal{T}_0^1 corresponds to the value trajectory that immediately increases at $t = 0$. Node 12 in \mathcal{T}_0^2 is the next to increase away from 0, and finally nodes 1 and 2 in \mathcal{T}_0^3 are the last of the nodes of \mathcal{S}_0 to increase from 0.



(a) LCP.



(b) ARC-P.

Figure 5: Malicious node attempts to drive the values of the normal nodes to a value of 2. The malicious node succeeds whenever LCP is used, but fails whenever ARC-P is used.

7. DISCUSSION

The notion of graph connectivity has long been the backbone of investigations into fault tolerant and secure distributed algorithms. Indeed, under the assumption of full knowledge of the network topology, connectivity is *the key* metric in determining whether a fixed number of malicious adversaries can be overcome. However, in large scale systems and complex networks, it is not practical for the various nodes to obtain knowledge of the global network topology. This necessitates the development of algorithms that allow the

Table 1: Related work for resilient consensus in synchronous networks using only local information (no nonlocal information, no relays, and the network is *not* complete).

Scope	Threat Model	
	Byzantine	Malicious
F -total	[17, 32]	[17], this paper
F -local	–	[34], this paper

nodes to operate on purely local information. This paper continues and extends the work started in [16, 17, 34], and represents a step in this direction for the particular application of distributed consensus. Using the ARC-P algorithm developed in [16], the notion of robust graphs introduced in [34], and the extensions of each presented here, we characterize necessary/sufficient conditions for the normal nodes in large-scale networks to mitigate the influence of adversaries. We show that the notions of robust digraphs are the appropriate analogues to graph connectivity when considering purely local filtering rules at each node in the network. Just as the notion of connectivity has played a central role in the existing analysis of reliable distributed algorithms with global topological knowledge, we believe that robust digraphs (and its variants) will play an important role in the investigation of purely local algorithms.

In a recent paper, developed independently of our work, Vaidya *et al.* have characterized the tight conditions for resilient consensus using only local information whenever the threat model is Byzantine and the scope of threat is F -total [32]. The network constructions used in [32] are very similar to the robust digraphs presented here. In particular, the networks in [32] also require redundancy of information flow between subsets of nodes in the network in a single hop.

Finally we summarize the main works related to resilient consensus using only local information in Table 1. In this table, we include only works on resilient consensus (also referred to as Byzantine approximate consensus, or just approximate consensus in the literature) in synchronous networks that use only local information, with no relaying of information across the network and with networks that are *not* complete (since complete networks provide global information and have high communication cost). Further discussion about the relationship of the results in this paper (and in [16, 17, 32, 34]) to approximate consensus can be found in [34] and [32].

8. ACKNOWLEDGMENTS

H. J. LeBlanc and X. Koutsoukos are supported in part by the National Science Foundation (CNS-1035655, CCF-0820088), the U.S. Army Research Office (ARO W911NF-10-1-0005), and Lockheed Martin. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government. H. Zhang and S. Sundaram are supported in part by a grant from the Natural Sciences and Engineering Research Council of Canada (NSERC), and by a grant from the Waterloo Institute for Complexity and Innovation (WICI).

9. REFERENCES

[1] N. Agmon and D. Peleg. Fault-tolerant gathering algorithms for autonomous mobile robots. *SIAM J. Comput.*, 36(1):56–82, 2006.

[2] R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74(1):47–97, Jan 2002.

[3] Z. Bouzid, M. G. Potop-Butucaru, and S. Tixeuil. Optimal byzantine-resilient convergence in uni-dimensional robot networks. *Theoretical Computer Science*, 411(34-36):3154–3168, 2010.

[4] A. A. Cárdenas, S. Amin, and S. Sastry. Research challenges for the security of control systems. In *Proceedings of the 3rd conference on Hot topics in security*, July 2008.

[5] D. W. Casbeer, D. B. Kingston, R. W. Beard, T. W. McLain, S.-M. Li, and R. Mehra. Cooperative forest fire surveillance using a team of small unmanned air vehicles. *International Journal of Systems Sciences*, 37(6):351–360, May 2006.

[6] N. Chopra and M. W. Spong. Passivity-based control of multi-agent systems. In *Advances in Robot Control, From Everyday Physics to Human-Like Movements*, Sadao Kawamura and Mikhail Svinin (Eds), pages 107 – 134. Springer Verlag, Berlin, 2006.

[7] J. Cortés. Distributed algorithms for reaching consensus on general functions. *Automatica*, 44(3):726 – 737, 2008.

[8] X. Défago, M. Gradinariu, S. Messika, and P. Raipin-Parvédy. Fault-tolerant and self-stabilizing mobile robots gathering. In S. Dolev, editor, *Distributed Computing*, volume 4167 of *Lecture Notes in Computer Science*, pages 46–60. Springer Berlin, Heidelberg, 2006.

[9] J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004.

[10] V. Gupta, C. Langbort, and R. Murray. On the robustness of distributed algorithms. In *IEEE Conf. on Decision and Control*, pages 3473 –3478, San Diego, California, December 2006.

[11] J. Hromkovic, R. Klasing, A. Pelc, P. Ruzicka, and W. Unger. *Dissemination of Information in Communication Networks*. Springer-Verlag, 2005.

[12] A. Ichimura and M. Shigeno. A new parameter for a broadcast algorithm with locally bounded Byzantine faults. *Information Processing Letters*, 110:514–517, 2010.

[13] A. Jadbabaie, J. Lin, and A. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Aut. Control*, 48(6):988 – 1001, jun. 2003.

[14] D. Kingston, R. Beard, and R. Holt. Decentralized perimeter surveillance using a team of UAVs. *IEEE Transactions on Robotics*, 24(6):1394 –1404, Dec. 2008.

[15] L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. *ACM Trans. Program. Lang. Syst.*, 4(2):382–401, 1982.

[16] H. J. LeBlanc and X. D. Koutsoukos. Consensus in networked multi-agent systems with adversaries. In *Proceedings of the 14th international conference on Hybrid systems: computation and control*, (HSCC ’11), pages 281–290, Chicago, IL, 2011.

[17] H. J. LeBlanc and X. D. Koutsoukos. Low complexity resilient consensus in networked multi-agent systems with adversaries. In *Proceedings of the 15th international conference on Hybrid systems: computation and control*, (HSCC ’12), Beijing, China, 2012. to appear.

[18] H. J. LeBlanc, H. Zhang, S. Sundaram, and X. D. Koutsoukos. Consensus of multi-agent networks in the presence of adversaries using only local information. *CoRR*, arxiv, 2012.

[19] J. Lorenz and D. Lorenz. On conditions for convergence to

- consensus. *IEEE Transactions on Automatic Control*, 55(7):1651–1656, July 2010.
- [20] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann Publishers Inc., San Francisco, California, 1997.
- [21] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Trans. on Aut. Control*, 50(2):169 – 182, Feb. 2005.
- [22] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.
- [23] F. Pasqualetti, A. Bicchi, and F. Bullo. Consensus computation in unreliable networks: A system theoretic approach. *IEEE Trans. on Aut. Control*, 57(1):90–104, Jan. 2012.
- [24] A. Pelc and D. Peleg. Broadcasting with locally bounded Byzantine faults. In *Information Processing Letters*, pages 109–115, 2005.
- [25] W. Ren and R. Beard. Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control*, 50(5):655 – 661, May 2005.
- [26] W. Ren, R. Beard, and E. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems Magazine*, 27(2):71–82, April 2007.
- [27] L. Scardovi and R. Sepulchre. Synchronization in networks of identical linear systems. *Automatica*, 45(11):2557–2562, 2009.
- [28] S. Sundaram and C. Hadjicostis. Distributed function calculation and consensus using linear iterative strategies. *IEEE Journal on Selected Areas in Communications*, 26(4):650–660, May 2008.
- [29] S. Sundaram and C. Hadjicostis. Distributed function calculation via linear iterative strategies in the presence of malicious agents. *IEEE Trans. on Aut. Control*, 56(7):1495–1508, July 2011.
- [30] B. Touri and A. Nedić. On ergodicity, infinite flow, and consensus in random models. *IEEE Transactions on Automatic Control*, 56(7):1593–1605, July 2011.
- [31] J. N. Tsitsiklis. *Problems in Decentralized Decision Making and Computation*. PhD thesis, Department of EECS, MIT, 1984.
- [32] N. Vaidya, L. Tseng, and G. Liang. Iterative approximate Byzantine consensus in arbitrary directed graphs. *CoRR*, abs/1201.4183, 2012.
- [33] L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems and Control Letters*, 53:65–78, 2004.
- [34] H. Zhang and S. Sundaram. Robustness of information diffusion algorithms to locally bounded adversaries. In *Proceedings of the American Control Conf.*, June 2012. to appear.