Consensus Computation in Unreliable Networks: A System Theoretic Approach

Fabio Pasqualetti Antonio Bicchi Francesco Bullo

Abstract

This work considers the problem of reaching consensus in an unreliable linear consensus network. A solution to this problem is relevant for several tasks in multi-agent systems including motion coordination, clock synchronization, and cooperative estimation. By modeling the unreliable nodes as unknown and unmeasurable inputs affecting the network, we recast the problem into an unknown-input system theoretic framework. Only relying on their direct measurements, the agents detect and identify the misbehaving agents using fault detection and isolation techniques. We consider both the case that misbehaviors are simply caused by faults, or that they are the product of a definite, malignant "Byzantine" strategy. We express the solvability conditions of the two cases in a system theoretic framework, and from a graph theoretic perspective. We show that generically any node can correctly detect and identify the misbehaving agents, provided that the connectivity of the network is sufficiently high. Precisely, for a linear consensus network to be generically resilient to k concurrent faults, the connectivity of the communication graph needs to be $2k+1$, if Byzantine agents are allowed, and $k+1$, if non-colluding agents are considered. We finally provide algorithms for detecting and isolating misbehaving agents. The first procedure applies standard fault detection techniques, and affords complete intrusion detection if global knowledge of the graph is available to each agent, at a high computational cost. The second method is designed to exploit the presence in a network of weakly interconnected subparts, and provides computationally efficient detection of misbehaving agents whose behavior deviates more than a threshold, which is quantified in terms of the interconnection structure.

I. Introduction

Distributed systems and networks have received much attention in the last years because of their flexibility and computational performance. One of the most frequent tasks to be accomplished by autonomous agents is to agree upon some parameters. Agreement variables represent quantities of interest such as the work load in a network of parallel computers, the clock speed for wireless sensor networks, the velocity, the rendezvous point, or the formation pattern for a team of autonomous vehicles; e.g., see [1], [2], [3].

Several algorithms achieving consensus have been proposed and studied in the computer science community [4]. In this work, we consider linear consensus iterations, where, at each time instant, each node updates its state as

This material is based upon work supported in part by the ARO Institute for Collaborative Biotechnology award DAAD19-03-D-0004, by the AFOSR MURI award FA9550-07-1-0528, by the Contract IST 224428 (2008) (STREP) "CHAT - Control of Heterogeneous Automation Systems: Technologies for scalability, reconfigurability and security," and by the CONET, the Cooperating Objects Network of Excellence, funded by the European Commission under FP7 with contract number FP7-2007-2-224053. The authors thank Dr. Natasha Neogi for insightful conversations, and the reviewers for their thoughtful and constructive remarks.

Fabio Pasqualetti is with the Center for Control, Dynamical Systems and Computation, University of California, Santa Barbara, fabiopas@engineering.ucsb.edu .

Antonio Bicchi is with the Centro I. R. "E. Piaggio", Università di Pisa, bicchi@ing.unipi.it .

Francesco Bullo is with the Center for Control, Dynamical Systems and Computation, University of California, Santa Barbara, bullo@engineering.ucsb.edu .

a weighted combination of its own value and those received from its neighbors [5], [6]. The choice of algorithm weights is a parameter that influences the convergence speed toward the steady state value [7].

Because of the lack of a centralized entity that monitors the activity of the nodes of the network, distributed systems are prone to attacks and components failure, and it is of increasing importance to guarantee trustworthy computation even in the presence of misbehaving parts [8]. Misbehaving agents can interfere with the nominal functions of the network in different ways. In this paper, we consider two extreme cases: that the deviations from their nominal behavior are due to genuine, random faults in the agents; or that agents can instead craft messages with the purpose of disrupting the network functions. In the first scenario, faulty agents are unaware of the structure and state of the network and ignore the presence of other faults. In the second scenario, the worst-case assumption is made that misbehaving agents have knowledge of the structure and state of the network, and may collude with others to produce the biggest damage. We refer to the first case as non-colluding, or faulty; to the second case as malicious, or Byzantine.

Reaching unanimity in an unreliable system is an important problem, well studied by computer scientists interested in distributed computing. A first characterization of the resilience of distributed systems to malicious attacks appears in [9], where the authors consider the task of agreeing upon a binary message sent by a "Byzantine general," when the communication graph is complete. In [10] the resilience of a partially connected[1] network seeking consensus is analyzed, and it is shown that the well-behaving agents of a network can always agree upon a parameter if and only if the number of malicious agents

(i) is less than $\frac{1}{2}$ of the network connectivity, and
(ii) is less than $\frac{1}{3}$ of the number of processors.

This result has to be regarded as a fundamental limitation of the ability of a distributed consensus system to sustain arbitrary malfunctioning: the presence of misbehaving Byzantine processors can be tolerated only if their number satisfies the above threshold, independently of whatever consensus protocol is adopted.

In this work, we consider linear consensus algorithms in which every agent, including the misbehaving ones, are assumed to send the same information to all their neighbors. This assumption appears to be realistic for most control scenarios. In a sensing network for instance, the data used in the consensus protocol consist of the measurements taken directly by the agents, and it is assumed that the measurements regarding the same quantity coincide. Also, in a broadcast network, the information is transmitted using broadcast messages, so that the content of a message is the same for all the receiving nodes. The problem of characterizing the resilience properties of linear consensus strategies has been partially addressed in recent works [11], [12], [13], where, for the malicious case, it is shown that, despite the limited abilities of the misbehaving agents, the resilience to external attacks is still limited by the connectivity of the network. In [11] the problem of detecting and identifying misbehaving agents in a linear consensus network is first introduced, and a solution is proposed for the single faulty agent case. In [12], [13], the authors provide one policy that k malicious agents can follow to prevent some of the nodes of a $2k$-connected network from computing the desired function of the initial state, or, equivalently, from reaching an agreement. On the contrary, if the connectivity is $2k+1$ or more, then the authors show that generically the set of misbehaving nodes is identified independent of its behavior, so that the desired consensus is eventually reached.

[1]The connectivity of a graph is the maximum number of disjoint paths between any two vertices of the graph. A graph is complete if it has connectivity $n-1$, where n is the number of vertices in the graph.

The main differences between this paper and the references [12], [13] are as follows. First, the method proposed in [12], [13] takes inspiration from parity space methods for fault detection, while, following our early work [11], we adopt here unknown-input observers techniques (cf. [14]). Second, we focus on consensus networks, and we derive specific results for this important case that cannot be assessed for general linear iterations. Third, we consider two different types of misbehaving agents, namely malicious and faulty agents, and we provide network resilience bounds for both cases. Fourth, we exhaustively characterize the complete set of policies that make a set of k agents undetectable and/or unidentifiable, as opposed to [12] where only a particular disrupting strategy is defined. Fifth, we study system theoretic properties of consensus systems (e.g., detectability, stabilizability, left-invertibility), and we quantify the effect of some misbehaving inputs on the network performance. Finally, we address the problem of detection complexity and we propose a computationally efficient detection method, as opposed to combinatorial procedures. Our approach also differs from the existing computer science literature, e.g., our analysis leads to the development of algorithms that can be easily extended to work on both discrete and continuous time linear consensus networks, and also with partial knowledge of the network topology.

The main contributions of this work are as follows. By recasting the problem of linear consensus computation in an unreliable system into a system theoretic framework, we provide alternative and constructive system-theoretic proofs of existing bounds on the number of identifiable misbehaving agents in a linear network, i.e., k Byzantine agents can be detected and identified if the network is $(2k+1)$-connected, and they cannot be identified if the network is $(2k)$-connected or less. Moreover, by showing some connections between linear consensus networks and linear dynamical systems, we exhaustively describe the strategies that misbehaving nodes can follow to disrupt a linear network that is not sufficiently connected. We prove that the inputs that allow the misbehaving agents to remain undetected or unidentified coincide with the inputs-zero of a linear system associated with the consensus network, and they can be ignored if only genuinely faulty agents are considered. For the latter case, we provide a novel and comprehensive analysis on the detection and identification of non-colluding agents. We show that k faulty agents can be identified if the network is $(k+1)$-connected, and cannot if the network is k-connected or less. For both the cases of Byzantine and non-colluding agents, we prove that the proposed bounds are generic with respect to the network communication weights, i.e., given an (unweighted) consensus graph, the bounds hold for almost all (consensus) choices of the communication weights. In other words, if we are given a $(k+1)$-connected consensus network for which k faulty agents cannot be identified, then a random and arbitrary small change of the communication weights (within the space of consensus weights) make the misbehaving agents identifiable with probability one. In the last part of the paper, we discuss the problem of detecting and identifying misbehaving agents when either the partial knowledge of the network or hardware limitations make it impossible to implement an exact identification procedure. We introduce a notion of network decentralization, in terms of relatively weakly connected subnetworks, and derive a sufficient condition on the consensus matrix that allows to identify a certain class of misbehaving agents under limited information on the network structure, and that ultimately leads to a prompt recovery of the network functionalities.

The rest of the paper is organized as follows. Section II briefly recalls some basic facts on the geometric approach to the study of linear systems, and on the fault detection and isolation problem. In Section III we model linear consensus networks with misbehaving agents. Section IV presents the conditions under which the misbehaving agents are detectable and identifiable. In Section V we characterize the effect of an unidentifiable attack on the network consensus state. In Section VI we show that the resilience of linear consensus networks to failures and external attacks is a generic property with respect to the consensus weights. In Section VII we present our algorithmic procedures. Precisely we derive an exact identification algorithm, and an approximate and low-complexity procedure. Finally, Sections VIII and IX contain respectively our numerical studies and our conclusion.

II. Notation and preliminary concepts

We adopt the same notation as in [15]. Let $n, m, p \in \mathbb{N}$, let $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{p\times n}$. Throughout the paper, let the triple (A, B, C) denote the linear discrete time system

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \tag{1}$$

and let the subspaces $\mathcal{B} \subseteq \mathbb{R}^{n\times n}$ and $\mathcal{C} \subseteq \mathbb{R}^{n\times n}$ denote respectively the image space $\mathrm{Im}(B)$ and the null space $\mathrm{Ker}(C)$. A subspace $\mathcal{V} \subseteq \mathbb{R}^{n\times n}$ is a $(A, \mathcal{B})$-controlled invariant if $A\mathcal{V} \subseteq \mathcal{V} + \mathcal{B}$, while a subspace $\mathcal{S} \subseteq \mathbb{R}^{n\times n}$ is a $(A, \mathcal{C})$-conditioned invariant if $A(\mathcal{S} \cap \mathcal{C}) \subseteq \mathcal{S}$. The set of all controlled invariants contained in $\mathcal{C}$ admits a supremum, which we denote with $\mathcal{V}^*$, and which corresponds to the locus of all possible state trajectories of (1) invisible at the output. On the other hand, the set of the conditioned invariants containing $\mathcal{B}$ admits an infimum, which we denote with $\mathcal{S}^*$. Several problems, including disturbance decoupling, non interacting control, fault detection and isolation, and state estimation in the presence of unknown inputs have been addressed and solved in a geometric framework [15], [16].

In the classical Fault Detection and Isolation (FDI) setup, the presence of sensor failures and actuator malfunctions is modeled by adding some unknown and unmeasurable functions $u_i(t)$ to the nominal system. The FDI problem is to design, for each failure i, a filter of the form

$$\begin{aligned} w_i(t+1) &= F_i w_i(t) + E_i y(t), \\ r_i(t) &= M_i w(t) + H_i y(t), \end{aligned} \tag{2}$$

also known as residual generator, that takes the observables $y(t)$ and generates a residual vector $r_i(t)$ that allows to uniquely identify if $u_i(t)$ becomes nonzero, i.e., if the failure i occurred in the system. Let $B_1, \dots, B_m$ be the input matrices of the failure functions $u_1, \dots, u_m$. As a result of [15], [17], the i-th failure can be correctly identified if and only if $\mathcal{B}_i \cap (\mathcal{V}^*_{K\setminus\{i\}} + \mathcal{S}^*_{K\setminus\{i\}}) = \emptyset$, where $\mathcal{V}^*_{K\setminus\{i\}}$ and $\mathcal{S}^*_{K\setminus\{i\}}$ are the maximal controlled and minimal conditioned invariant subspaces associated with the triple $(A, [B_1 \cdots B_{i-1} \;\; B_{i+1} \cdots B_m], C)$. It can be shown that, under the above solvability condition, the filter (2) can be designed as a dead beat device to have finite convergence time [17]: this property will be used in Section VII for the characterization of our intrusion detection algorithm. We remark that, although the FDI problem does not coincide with the problem we are going to face, we will be using some standard FDI techniques to design our identification algorithms, and we refer the reader to [14] for a comprehensive treatment of the subject.

III. Linear consensus in the presence of misbehaving agents

Let G denote a directed graph with vertex set $V = \{1, \dots, n\}$ and edge set $E \subset V \times V$, and recall that the connectivity of G is the maximum number of disjoint paths between any two vertices of the graph, or, equivalently, the minimum number of vertices in a vertex cutset [18]. The neighbor set of a node $i \in V$, i.e., all the nodes $j \in V$ such that the pair $(j, i) \in E$, is denoted with N_i. We let each vertex $j \in V$ denote an autonomous agent, and we associate a real number x_j with each agent j. Let the vector $x \in \mathbb{R}^n$ contain the values x_j. A linear iteration over G is an update rule for x and is described by the linear discrete time system

$$x(t+1) = Ax(t), \tag{3}$$

where the (i,j)-th entry of A is nonzero only if $(j,i) \in E$. If the matrix A is row stochastic and primitive, then, independent of the initial values of the nodes, the network asymptotically converges to a configuration in which the state of the agents coincides. In the latter case, the matrix A is referred to as a consensus matrix, and the system (3) is called *consensus* system. The graph G is referred to as the communication graph associated with the consensus system (3) or, equivalently, with the consensus matrix A. A detailed treatment of the applications, and the convergence aspects of the consensus algorithm is in [1], [2], [3], and in the references therein.

We allow for some agents to update their state differently than specified by the matrix A by adding an exogenous input to the consensus system. Let $u_i(t)$, $i \in V$, be the input associated with the i-th agent, and let $u(t)$ be the vector of the functions $u_i(t)$, then the consensus system becomes $x(t+1) = Ax(t) + u(t)$.

Definition 1 (Misbehaving agent) *An agent j is* misbehaving *if there exists a time $t \in \mathbb{N}$ such that $u_j(t) \neq 0$.*

In Section IV we will give a precise definition of the distinction, made already in the Introduction, between *faulty* and *malicious* agents, on the basis of their inputs.

Let $K = \{i_1, i_2, \dots\} \subseteq V$ denote a set of misbehaving agents, and let $B_K = [e_{i_1} \ \ e_{i_2} \ \ \cdots]$, where e_i is the i-th vector of the canonical basis. The consensus system with misbehaving agents K assumes the form

$$x(t+1) = Ax(t) + B_K u_K(t). \tag{4}$$

As it is shown in [11], algorithms of the form (3) have no resilience to malfunctions, and the presence of a misbehaving agent may prevent the entire network from reaching consensus. As an example, let $c \in \mathbb{R}$, and let $u_i(t) = -A_i x(t) + c$, where A_i denotes the i-th row of A. After reordering the variables in a way that the well-behaving nodes come first, the consensus system can be rewritten as

$$\tilde{x}(t+1) = \begin{bmatrix} Q & R \\ 0 & 1 \end{bmatrix} \tilde{x}(t), \tag{5}$$

where the matrix Q corresponds to the interaction among the nodes $V \setminus \{i\}$, while R denotes the connection between the sets $V \setminus \{i\}$ and $\{i\}$. Recall that a matrix is said to be Schur stable if all its eigenvalues lie in the open unit disk.

Lemma III.1 (Quasi-stochastic submatrices) *Let A be an $n \times n$ consensus matrix, and let J be a proper subset of $\{1, \dots, n\}$. The submatrix with entries $A_{i,k}$, $i, k \in J$, is Schur stable.*

Proof: Reorder the nodes such that the indexes in J come first in the matrix A. Let A_J be the leading principal submatrix of dimension $|J|$. Let $\tilde{A}_J = \left[\begin{smallmatrix} A_J & 0 \\ 0 & 0 \end{smallmatrix}\right]$, where the zeros are such that $\tilde{A}_J$ is $n \times n$, and note that $\rho(A_J) = \rho(\tilde{A}_J)$, where $\rho(A_J)$ denotes the spectral radius of the matrix A_J [19]. Since A is a consensus matrix, it has only one eigenvalue of unitary modulus, and $\rho(A) = 1$. Moreover, $A \geq |\tilde{A}_J|$, and $A \neq |\tilde{A}_J|$, where $|\tilde{A}_J|$ is such that its (i,j)-th entry equals the absolute value of the (i,j)-th entry of $\tilde{A}_J$, $\forall i, j$. It is known that $\rho(A_J) \leq \rho(A) = 1$, and that if equality holds, then there exists a diagonal matrix D with nonzero diagonal entries, such that $A = D\tilde{A}_J D^{-1}$ [20]. Because A is irreducible, there exists no diagonal matrix D with nonzero diagonal entries such that $A = D\tilde{A}_J D^{-1}$. We conclude that $\rho(A_J) < \rho(A) = 1$. ∎

Because of Lemma III.1, the matrix Q in (5) is Schur stable, so that the steady state value of the well-behaving agents in (5) depends upon the action of the misbehaving node, and it corresponds to $(I-Q)^{-1}Rc$. In particular, since $(I-Q)^{-1}R = [1 \cdots 1]^\mathsf{T}$, a single misbehaving agent can steer the network towards any consensus value by choosing the constant c.[2]

It should be noticed that a different model for the misbehaving nodes consists in the modification of the entries of A corresponding to their incoming communication edges. However, since the resulting network evolution can be obtained by properly choosing the input $u_K(t)$ and letting the matrix A fixed, our model does not limit generality, while being convenient for the analysis. For the same reason, system (4) also models the case of defective communication edges. Indeed, if the edge from the node i to the node j is defective, then the message received by the agent j at time t is incorrect, and hence also the state $x_j(\bar{t})$, $\bar{t} \geq t$. Since the values $x_j(\bar{t})$ can be produced with an input $u_j(t)$, the failure of the edge (i,j) can be regarded as the j-th misbehaving action. Finally, the following key difference between our model and the setup in [10] should be noticed. If the communication graph is complete, then up to $n-1$ (instead of $\lfloor n/3 \rfloor$) misbehaving agents can be identified in our model by a well-behaving agent. Indeed, since with a complete communication graph the initial state $x(0)$ is correctly received by every node, the consensus value is computed after one communication round, so that the misbehaving agents cannot influence the dynamics of the network.

IV. Detection and identification of misbehaving agents

The problem of ensuring trustworthy computation among the agents of a network can be divided into a detection phase, in which the presence of misbehaving agents is revealed, and an identification phase, in which the identity of the intruders is discovered. A set of misbehaving agents may remain undetected from the observations of a node j if there exists a normal operating condition under which the node would receive the same information as under the perturbation due to the misbehavior. To be more precise, let $C_j = [e_{n_1} \ \ldots \ e_{n_p}]^\mathsf{T}$, $\{n_1, \ldots, n_p\} = N_j$, denote the output matrix associated with the agent j, and let $y_j(t) = C_j x(t)$ denote the measurements vector of the j-th agent at time t. Let $x(x_0, \bar{u}, t)$ denote the network state trajectory generated from the initial state x_0 under the input sequence $\bar{u}(t)$, and let $y_j(x_0, \bar{u}, t)$ be the sequence measured by the j-th node and corresponding to the same initial condition and input.

Definition 2 (Undetectable input) *For a linear consensus system of the form* (4)*, the input* $u_K(t)$ *introduced by a set* K *of misbehaving agents is* undetectable *if*

$$\exists\, x_1, x_2 \in \mathbb{R}^n, j \in V : \forall t \in \mathbb{N}, y_j(x_1, u_K, t) = y_j(x_2, 0, t).$$

A more general concern than detection is identifiability of intruders, i.e. the possibility to distinguish from measurements between the misbehaviors of two distinct agents, or, more generally, between two disjoint subsets of agents. Let $\mathcal{K} \subset 2^V$ contain all possible sets of misbehaving agents.[3]

[2] If the misbehaving input is not constant, then the network may not achieve consensus. In particular, the effect of a misbehaving input u_K on the network state at time t is given by $\sum_{\tau=0}^{t} A^{t-\tau} B_K u_K(\tau)$ (see also Section V).

[3] An element of $\mathcal{K}$ is a subset of $\{1, \ldots, n\}$. For instance, $\mathcal{K}$ may contain all the subsets of $\{1, \ldots, n\}$ with a specific cardinality.

Definition 3 (Unidentifiable input) *For a linear consensus system of the form* (4) *and a nonempty set* $K_1 \in \mathcal{K}$, *an input* $u_{K_1}(t)$ *is* unidentifiable *if there exist* $K_2 \in \mathcal{K}$, *with* $K_1 \cap K_2 = \emptyset$, *and an input* $u_{K_2}(t)$ *such that*

$$\exists\, x_1, x_2 \in \mathbb{R}^n, j \in V : \forall t \in \mathbb{N}, y_j(x_1, u_{K_1}, t) = y_j(x_2, u_{K_2}, t).$$

Of course, an undetectable input is also unidentifiable, since it cannot be distinguished from the zero input. The converse does not hold. Unidentifiable inputs are a very specific class of inputs, to be precisely characterized later in this section. Correspondingly, we define

Definition 4 (Malicious behaviors) *A set of misbehaving agents K is malicious if its input $u_K(t)$ is unidentifiable. It is faulty otherwise.*

We provide now a characterization of malicious behaviors for the particularly important class of linear consensus networks. Notice however that, if the matrix A below is not restricted to be a consensus matrix, then the following Theorem extends the results in [12] by fully characterizing the inputs for which a group of misbehaving agents remains unidentified from the output observations of a certain node.

Theorem IV.1 (Characterization of malicious behaviors) *For a linear consensus system of the form* (4) *and a nonempty set* $K_1 \in \mathcal{K}$, *an input* $u_{K_1}(t)$ *is unidentifiable if and only if*

$$\sum_{\tau=0}^{t} C_j A^{t-\tau} B_{K_1} u_{K_1}(\tau) = C_j A^{t+1} \bar{x} + \sum_{\tau=0}^{t} C_j A^{t-\tau} B_{K_2} u_{K_2}(\tau),$$

for all $t \in \mathbb{N}$, *and for some* $u_{K_2}(t)$, *with* $K_2 \in \mathcal{K}$ *and* $K_1 \cap K_2 = \emptyset$, *and* $\bar{x} \in \mathbb{R}^n$. *If the same holds with* $u_{K_2}(t) \equiv 0$, *the input is actually undetectable.*

Proof: By definitions 2 and 3, an input $u_{K_1}(t)$ is unidentifiable if $y_j(x_1, u_{K_1}, t) = y_j(x_2, u_{K_2}, t)$, and it is undetectable if $y_j(x_1, u_{K_1}, t) = y_j(x_2, 0, t)$, for some x_1, x_2, and $u_{K_2}(t)$. Because of the linearity of the network, the statement follows. ■

Remark 1 (Malicious behaviors are not generic) Because an unidentifiable input must satisfy the equation in Theorem IV.1, excluding pathological cases, unidentifiable signals are not generic, and they can be injected only intentionally by colluding misbehaving agents. This motivates our definition of "malicious" for those agents which use unidentifiable inputs.

We consider now the resilience of a consensus network to faulty and malicious misbehaviors. Let I denote the identity matrix of appropriate dimensions. The zero dynamics of the linear system (A, B_K, C_j) are the (nontrivial) state trajectories invisible at the output, and can be characterized by means of the $(n + p) \times (n + m)$ pencil

$$P(z) = \begin{bmatrix} zI - A & B_K \\ C_j & 0 \end{bmatrix}.$$

The complex value $\bar{z}$ is said to be an invariant zero of the system (A, B_K, C_j) if there exists a state-zero direction x_0, and an input-zero direction g, such that $(\bar{z}I - A)x_0 + B_K g = 0$, and $C_j x_0 = 0$. Also, if *rank*$(P(z)) = n+m$ for all but finitely many complex values z, then the system (A, B_K, C_j) is left-invertible, i.e., starting from any initial condition, there are no two distinct inputs that give rise to the same output sequence [21]. We next characterize the relationship between the zero dynamics of a consensus system and the connectivity of the consensus graph.

Lemma IV.1 (Zero dynamics and connectivity) *Given a k-connected linear network with matrix A, there exists a set of agents K_1, with $|K_1| > k$, and a node j such that the consensus system (A, B_{K_1}, C_j) is not left-invertible. Furthermore, there exists a set of agents K_2, with $|K_2| = k$, and a node j such that the system (A, B_{K_2}, C_j) has nontrivial zero dynamics.*

Proof: Let G be the digraph associated with A, and let k be the connectivity of G. Take a set K of $k+1$ misbehaving nodes, such that k of them form a vertex cut S of G. Note that, since the connectivity of G is k, such a set always exists. The network G is divided into two subnetworks G_1 and G_3, which communicate only through the nodes S. Assume that the misbehaving agent $K \setminus S$ belongs to G_3, while the observing node j belongs to G_1. After reordering the nodes such that the vertices of G_1 come first, the vertices S come second, and the vertices of G_3 come third, the consensus matrix A is of the form $\begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{21} & A_{22} & A_{23} \\ 0 & A_{32} & A_{33} \end{bmatrix}$, where the zero matrices are due to the fact that S is a vertex cut. Let $u_S(t) = -A_{23}x_3(t)$, where x_3 is the vector containing the values of the nodes of G_3, and let $u_{K\setminus S}(t)$ be any arbitrary nonzero function. Clearly, starting from the zero state, the values of the nodes of G_1 are constantly 0, while the subnetwork G_3 is driven by the misbehaving agent $K \setminus S$. We conclude that the triple (A, B_K, C_j) is not left-invertible.

Suppose now that $K \equiv S$ as previously defined, and let $u_K(t) = -A_{23}x_3(t)$. Let the initial condition of the nodes of G_1 and of S be zero. Since every state trajectory generated by $x_3 \neq 0$ does not appear in the output of the agent j, the triple (A, B_K, C_j) has nontrivial zero dynamics. ■

It is straightforward to show that a misbehaving set K is detectable by the agent j if and only if the system (A, B_K, C_j) has no zero dynamics. Hence, following Lemma IV.1, we next state an upper bound on the number of misbehaving agents that can be detected.

Theorem IV.2 (Detection bound) *Given a k-connected linear consensus network, there exist undetectable inputs for a specific set of k misbehaving agents.*

Proof: Let K , with $|K| = k$, be the misbehaving set, and let K form a vertex cut of the consensus network. Because of Lemma IV.1, for some output matrix C_j, the consensus system has nontrivial zero dynamics, i.e., there exists an initial condition $x(0)$ and an input $u_K(t)$ such that $y_j(t) = 0$ at all times. Hence, the input $u_K(t)$ is undetectable from the observations of j. ■

We now consider the identification problem.

Theorem IV.3 (Identification of misbehaving agents) *For a set of misbehaving agents $K_1 \in \mathcal{K}$, every input is identifiable from j if and only if the consensus system $(A, [B_{K_1} \;\; B_{K_2}], C_j)$ has no zero dynamics for every $K_2 \in \mathcal{K}$.*

Proof: (Only if) By contradiction, let x_0 and $[u_{K_1}^\mathsf{T} \;\; -u_{K_2}^\mathsf{T}]^\mathsf{T}$ be a state-zero direction, and an input-zero sequence for the system $(A, [B_{K_1} \;\; B_{K_2}], C_j)$. We have

$$y_j(t) = 0 = C_j\left(A^t x_0 + \sum_{\tau=0}^{t-1} A^{t-\tau-1} B_{K_1} u_{K_1}(\tau) - \sum_{\tau=0}^{t-1} A^{t-\tau-1} B_{K_2} u_{K_2}(\tau)\right),$$

Therefore,

$$C_j\left(A^t x_0^1 + \sum_{\tau=0}^{t-1} A^{t-\tau-1} B_{K_1} u_{K_1}(\tau)\right) = C_j\left(A^t x_0^2 + \sum_{\tau=0}^{t-1} A^{t-\tau-1} B_{K_2} u_{K_2}(\tau)\right),$$

where $x_0^1 - x_0^2 = x_0$. Clearly, since the output sequence generated by K_1 coincide with the output sequence generated by K_2, the two inputs are unidentifiable.

(If) Suppose that, for any $K_2 \in \mathcal{K}$, the system $(A[B_{K_1}\ B_{K_2}])$ has no zero dynamics, i.e., there exists no initial condition x_0 and input $[u_{K_1}^\mathsf{T}\ u_{K_2}^\mathsf{T}]^\mathsf{T}$ that result in the output being zero at all times. By the linearity of the network, every input u_{K_1} is identifiable. ■

As a consequence of Theorem IV.3, if up to k misbehaving agents are allowed to act in the network, then a necessary and sufficient condition to correctly identify the set of misbehaving nodes is that the consensus system subject to any set of $2k$ inputs has no nontrivial zero dynamics.

Theorem IV.4 (Identification bound) *Given a k-connected linear consensus network, there exist unidentifiable inputs for a specific set of $\lfloor \frac{k-1}{2} \rfloor + 1$ misbehaving agents.*

Proof: Since $2(\lfloor \frac{k-1}{2} \rfloor + 1) \geq k$, by Lemma IV.1 there exist K_1, K_2, with $|K_1| = |K_2| = \lfloor \frac{k-1}{2} \rfloor + 1$, and j such that the system $(A, [B_{K_1}\ \ B_{K_2}], C_j)$ has nontrivial zero dynamics. By Theorem IV.3, there exists an input and an initial condition such that the set K_1 is undistinguishable from K_2 to the agent j. ■

So far we have shown that, in the worst case, in a k-connected network, at most $k-1$ (resp. $\lfloor \frac{k-1}{2} \rfloor$) misbehaving agents can be detected (resp. identified) by every agent. Notice that, for a linear consensus network, Theorem IV.4 provides an alternative proof of the resilience bound first presented in [10] and in [12].

We now focus on the faulty misbehavior case. Notice that, because such agents are identifiable by definition, we only need to guarantee the existence of identifiable inputs. We start by showing that, independent of the cardinality of a set K, there exist detectable inputs for a consensus system (A, B_K, C_j), so that any set of faulty agents is detectable. By using a result from [22], an input $u_K(t)$ is undetectable from the measurements of the j-th agent only if for all $t \in \mathbb{N}$, it holds $C_j A^v B_K u_K(t) = C_j A^{v+1} x(t)$, where $C_j A^v B_K$ is the first nonzero Markov parameter, and $x(t)$ is the network state at time t. Notice that, because of the irreducibility assumption of a consensus matrix, independently of the cardinality of the faulty set and of the observing node j, there always exists a finite v such that $C_j A^v B_K \neq 0$, so that every input $u_K(t) \neq (C_j A^v B_K)^\dagger C_j A^{v+1} x(t)$ is detectable. We now characterize an upper bound on the maximum number of identifiable inputs, or, equivalently, on the number of identifiable faulty agents. We show that, if the number of misbehaving components is allowed to equal the connectivity of consensus network, then there exists a set of misbehaving agents that are unidentifiable independent of their input.

Theorem IV.5 (Identification of faulty agents) *Given a k-connected linear consensus network, there exists no identifiable input for a specific set of k misbehaving agents*

Proof: Let K_1, with $|K_1| = k$, form a vertex cut. The network is divided into two subnetworks G_1 and G_2 by the agents K_1. Let K_2, with $|K_2| \leq k$, be the set of faulty agents, and suppose that the set K_2 belongs to the subnetwork G_2. Let j be an agent of G_1. Notice that, because K_1 forms a vertex cut, for every initial condition $x(0)$ and for every input $u_{K_2}(t)$, there exists an input $u_{K_1}(t)$ such that the output sequences at the node j coincide. In other words, every input $u_{K_2}(t)$ is unidentifiable. ■

Hence, in a k-connected network, a set of k faulty agents may remain unidentified independent of its input function. It should be noticed that Theorems IV.4 and IV.5 only give an upper bound on the maximum number of concurrent misbehaving agents that can be detected and identified. In Section VI it will be shown that, generically, in a k-connected network, there exists no unidentifiable input for any set of $\lfloor \frac{k-1}{2} \rfloor$ misbehaving agents, and there exist identifiable inputs for any set of $k-1$ misbehaving agents. In other words, if there exists a set of misbehaving nodes that cannot be identified by an agent, then, provided that the connectivity of the communication graph is sufficiently high, a random and arbitrarily small change of the consensus matrix makes the misbehaving nodes detectable and identifiable with probability one.

V. Effects of unidentified misbehaving agents

In the previous section, the importance of zero dynamics in the misbehavior detection and identification problem has been shown. In particular, we proved that a misbehaving agent may alter the nominal network behavior while remaining undetected by injecting an input-zero associated with the current network state. We now study the effect of an unidentifiable attack on the final consensus value. As a preliminary result, we prove the detectability of a consensus network.

Lemma V.1 (Detectability) *Let the matrix A be row stochastic and irreducible.*[4] *For any network node j, the pair (A, C_j) is detectable.*

Proof: If A is stochastic and irreducible, then it has at least $h \geq 1$ eigenvalues of unitary modulus. Precisely, the spectrum of A contains $\{1 = e^{i\theta_0}, e^{i\theta_1}, \dots, e^{i\theta_{h-1}}\}$. By Wielandt's theorem [19], we have $AD_k = e^{i\theta_k} D_k A$, where $k \in \{0, \dots, h-1\}$, and D_k is a full rank diagonal matrix. By multiplying both sides of the equality by the vector of all ones, we have $AD_k\mathbf{1} = e^{i\theta_k} D_k A\mathbf{1} = e^{i\theta_k} D_k\mathbf{1}$, so that $D_k\mathbf{1}$ is the eigenvector associated with the eigenvalue $e^{i\theta_k}$. Observe that the vector $D_k\mathbf{1}$ has no zero component, and that, by the eigenvector test [21], the pair (A, C_j) is detectable. Indeed, since A is irreducible, the neighbor set N_j is nonempty, and the eigenvector $D_k\mathbf{1}$, with $k \in \{0, \dots, h-1\}$, is not contained in $\mathrm{Ker}(C_j)$. ■

By duality, a result on the stabilizability of the pair (A, B_j) can also be asserted.

Lemma V.2 (Stabilizability) *Let the matrix A be row stochastic and irreducible. For any network node j, the pair (A, B_j) is stabilizable.*

Remark 2 (State estimation via local computation) If a linear system is detectable (resp. stabilizable), then a linear observer (resp. controller) exists to asymptotically estimate (resp. stabilize) the system state. By combining the above results with Lemma III.1, we have that, under a mild assumption on the matrix A, the state of a linear network can be asymptotically observed (resp. stabilized) via local computation. Consider for instance the problem of designing an observer [15], and let $C_j = e_j^{\mathsf{T}}$. Take $G = -A_j$, where A_j denotes the j-th column of A. By permuting the entries, the matrix $A + GC_j$ can be written as a block-triangular matrix, i.e.,

$$A + GC_j = \begin{bmatrix} * & 0 \\ * & 0 \end{bmatrix},$$

which is stable because of Lemma III.1. Finally, since the nonzero entries of G correspond to the out-neighbors[5] of the node j, the output injection operation GC_j only requires local information.

[4] Notice that the primitivity of A is not assumed here.

[5] The agent i is an out-neighbor of j if the (i, j)-th entry of A is nonzero, or, equivalently, if (j, i) is an edge.

A class of undetectable attacks is now presented. Notice that misbehaving agents can arbitrarily change their initial state without being detected during the consensus iterations, and, by doing so, misbehaving components can cause at most a constant error on the final consensus value. Indeed, let A be a consensus matrix, and let K be the set of misbehaving agents. Let $x(0)$ be the network initial state, and suppose that the agents K alter their initial value, so that the network initial state becomes $x(0) + B_K c$, where $c \in \mathbb{R}^{|K|}$. Recall from [19] that $\lim_{t\to\infty} A^t = \mathbf{1}\pi$, where $\mathbf{1}$ is the vector of all ones, and π is such that $\pi A = \pi$. Therefore , the effect of the misbehaving set K on the final consensus state is $\mathbf{1}\pi B_K c$. Clearly, if the vector $x(0)+B_K c$ is a valid initial state, the misbehaving agents cannot be detected. On the other hand, since it is possible for uncompromised nodes to estimate the observable part of the initial state of the whole network, if an acceptability region (or an a priori probability distribution) is available on initial states, then, by analyzing the reconstructed state, a form of intrusion detection can be applied, e.g., see [23]. We conclude this paragraph by showing that misbehaving agents have no interest in altering their initial states while remaining completely undetected by an observer, since, by doing so, they cannot alter the final consensus value. Indeed, for the attack to be undetectable from an agent j, the vector $B_K c$ needs to belong to the unobservable subspace of (A, C_j). Let v be an eigenvector associated with the unobservable eigenvalue $\bar{z}$, i.e., $(\bar{z}I - A)v = 0$ and $C_j v = 0$. We have $\pi(\bar{z}I - A)v = (\bar{z} - 1)\pi v = 0$, and, because of the detectability of (A, C_j), it follows $|\bar{z}| < 1$ (cf. Lemma V.1), and hence $\pi v = 0$. Therefore, if the attack $B_K c$ is undetectable from any of the agents, then $\lim_{t\to\infty} A^t B_K c = \mathbf{1}\pi B_K c = 0$, so that the change of the initial states of misbehaving agents does not affect the final consensus value.

A different class of unidentifiable attacks consists of injecting a signal corresponding to an input-zero for the current network state. We start by characterizing the potential disruption caused by misbehaving nodes that introduce nonzero, but exponentially vanishing inputs.[6]

Lemma V.3 (Exponentially stable input) *Let A be a consensus matrix, and let K be a set of agents. Let $u : \mathbb{N} \mapsto \mathbb{R}^{|K|}$ be exponentially decaying. There exists $z \in (0,1)$ and $\bar{u} \in \mathbb{R}^{|K|}$ such that*

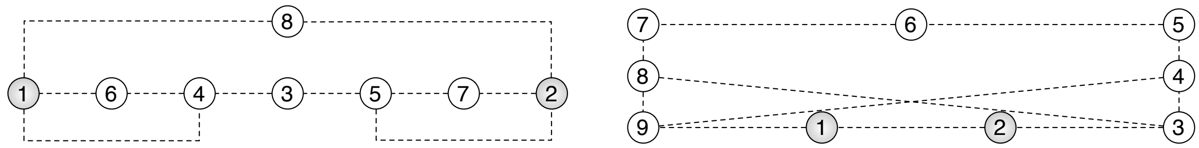
$$\lim_{t\to\infty} \sum_{\tau=0}^{t} A^{t-\tau} B_K u(\tau) \preceq (1-z)^{-1} \mathbf{1} \pi B_K \bar{u},$$

where $\preceq$ denotes componentwise inequality, $\mathbf{1}$ is the vector of all ones of appropriate dimension, and π is such that $\pi A = \pi$.

Proof: Let $z \in (0,1)$ and $0 \preceq u_0 \in \mathbb{R}^{|K|}$ be such that $u(k) \preceq z^k u_0$. Then, since A is a nonnegative matrix, for all $t, \tau \in \mathbb{N}$, with $t \geq \tau$, we have $A^{t-\tau} B_K u(\tau) \preceq A^{t-\tau} B_K z^\tau u_0$, and hence $\lim_{t\to\infty} \sum_{\tau=0}^{t} A^{t-\tau} B_K u(\tau) \preceq \lim_{t\to\infty} \sum_{\tau=0}^{t} A^{t-\tau} B_K z^\tau u_0$. Notice that $(1-z)^{-1} = \lim_{t\to\infty} \sum_{\tau=0}^{t} z^\tau$. We now show that $\lim_{t\to\infty} \sum_{\tau=0}^{t} z^\tau (\mathbf{1}\pi - A^{t-\tau}) = \lim_{t\to\infty} \sum_{\tau=0}^{t} E(t,\tau) \preceq 0$, from which the theorem follows. Let $e(t,\tau)$ be any component of $E(t,\tau)$. Because $\lim_{t\to\infty} A^t = \mathbf{1}\pi$, there exist c and ρ, with $|z| \leq |\rho| < 1$, such that $e(t,\tau) \leq c z^\tau \rho^{t-\tau}$. We have

$$\lim_{t\to\infty} \sum_{\tau=0}^{t} c z^\tau \rho^{t-\tau} = \lim_{t\to\infty} c\rho^t \sum_{\tau=0}^{t} z^\tau \rho^{-\tau} = 0,$$

[6]An output-zeroing input can always be written as $u(k) = -(CA^\nu B)^\dagger CA^{\nu+1}(K_\nu A)^k x(0) - (CA^\nu B)^\dagger CA^{\nu+1} \left(\sum_{l=0}^{k-1} (K_\nu A)^{k-1-l} B u_h(l)\right) + u_h(h)$, where $\nu \in \mathbb{N}$, $(CA^\nu B)$ is the first nonzero Markov parameter, $K_\nu = I - B(CA^\nu B)^\dagger CA^\nu$ is a projection matrix, $x(0) \in \bigcap_{l=0}^{\nu} \mathrm{Ker}(CA^l)$ is the system initial state, and $u_h(k)$ is such that $CA^\nu B u_h(k) = 0$ [22].



(a) (b)

Fig. 1. In Fig. 1(a) The agents $\{1,2\}$ are misbehaving. The consensus system $(A, B_{\{1,2\}}, C_3)$ has unstable zeros. In Fig. 1(b) the agents $\{1,2\}$ are misbehaving. The consensus system $(A, B_{\{1,2\}}, C_6)$ is not left-invertible.

so that $\sum_{\tau=0}^{t} E(t,\tau)$ converges to zero as t approaches infinity. ■

Following Lemma V.3, if the zero dynamics are exponentially stable, then misbehaving agents can affect the final consensus value by a constant amount without being detected, if and only if they inject vanishing inputs along input-zero directions. If an admissible region is known for the network state, then a tight bound on the effect of misbehaving agents injecting vanishing inputs can be provided. Notice moreover that, in this situation, a well-behaving agent is able to detect misbehaving agents whose state is outside an admissible region by simply analyzing its state. Finally, for certain consensus networks, the effect of an exponentially stable input decreases to zero with the cardinality of the network. Indeed, let $\pi = \bar{\pi}/n$, where $\bar{\pi}$ is a constant row vector and n denotes the cardinality of the network.[7] Then, when n grows, the effect of the input $u(t) = z^t \bar{u}$, with $|z| < 1$, on the consensus value becomes negligible.

The left-invertibility and the stability of the zero dynamics is not an inherent property of a consensus system. Consider for instance the graph of Fig. 1(a), where the agents $\{1,2\}$ are malicious. If the network matrices are

$$A = \begin{bmatrix} 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\ 1/16 & 0 & 5/8 & 1/16 & 0 & 1/4 & 0 & 0 \\ 0 & 1/16 & 1/4 & 0 & 5/16 & 0 & 3/8 & 0 \\ 1/2 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 0 & 0 & 2/3 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B_{\{1,2\}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad C_3 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix},$$

then the system $(A, B_{\{1,2\}}, C_3)$ is left-invertible, but the invariant zeros are $\{0, +2, -2\}$. Hence, for given initial conditions, there exist non vanishing input sequences that do not appear in the output. Moreover, for the graph in Fig. 1(b), let the network matrices be

$$A = \begin{bmatrix} 1/3 & 1/3 & 0 & 0 & 0 & 0 & 0 & 0 & 1/3 \\ 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/4 & 1/4 & 1/4 & 0 & 0 & 0 & 1/4 & 0 \\ 0 & 0 & 1/4 & 1/4 & 1/4 & 0 & 0 & 0 & 1/4 \\ 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 1/4 & 0 & 0 & 0 & 1/4 & 1/4 & 1/4 \\ 1/4 & 0 & 0 & 1/4 & 0 & 0 & 0 & 1/4 & 1/4 \end{bmatrix}, \quad B_{\{1,2\}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad C_6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

It can be verified that the system $(A, B_{\{1,2\}}, C_6)$ is not left-invertible. Indeed, for zero initial conditions, any input of the form $u_1 = -u_2$ does not appear in the output sequence of the agent 6. In some cases, the left-invertibility of a consensus system can be asserted independently of the consensus matrix.

Theorem V.1 (Left-invertibility, single intruder case) *Let A be a consensus matrix, and let $B_i = e_i$, $C_j = e_j^{\mathsf{T}}$. Then the system (A, B_i, C_j) is left-invertible.*

[7]For instance, if A is doubly stochastic, then $\bar{\pi} = \mathbf{1}^{\mathsf{T}}$ [19].

Proof: Suppose, by contradiction, that (A, B_i, C_j) is not left-invertible. Then there exist state trajectories that, starting from the origin, are invisible to the output. In other words, since the input is a scalar, the Markov parameters $C_j A^t B_i$ have to be zero for all t. Notice the (i,k)-th component of A^t is nonzero if there exists a path of length t from i to k. Because A is irreducible, there exists t such that $C_j A^t B_i \neq 0$, and therefore the consensus system is left-invertible. ■

If in Theorem V.1 one identifies the i-th node with a single intruder, and the j-th node with an observer node, the theorem states that, for known initial conditions of the network, any two distinct inputs generated by a single intruder produce different outputs at all observing nodes, and hence can be detected. Consider for example a flocking application, in which the agent are supposed to agree on the velocity to be maintained during the execution of the task [1]. Suppose that a linear consensus iteration is used to compute a common velocity vector, and suppose that the algorithm has terminated, so that the states of the agents are equal to each other. Then no single misbehaving agent can change the velocity of the team without being detected, because no zero dynamic can be generated by a single agent starting from a consensus state.

We now consider the case in which several misbehaving agents are allowed to act simultaneously. The following result relating the position of the misbehaving agents in the network and the zero dynamics of a consensus system can be asserted.

Theorem V.2 (Stability of zero dynamics) *Let K be a set of agents and let j be a network node. The zero dynamics of the consensus system (A, B_K, C_j) are exponentially stable if one of the following is true:*

(i) *the system (A, B_K, C_j) is left-invertible, and there are no edges from the nodes K to $V \setminus \{N_j \cup K\}$;*
(ii) *the system (A, B_K, C_j) is left-invertible, and there are no edges from the nodes $V \setminus \{N_j \cup K\}$ to N_j; or*
(iii) *the sets K and N_j are such that $K \subseteq N_j$.*

Proof: Let z be an invariant zero, x and u a state-zero and input-zero direction, so that

$$(zI - A)x + B_K u = 0, \text{ and } C_j x = 0 \tag{6}$$

Reorder the nodes such that the set K comes first, the set $N_j \setminus K$ second, and the set $V \setminus \{K \cup N_j\}$ third. The consensus matrix and the vector x are accordingly partitioned as

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix},$$

and the input and output matrices become $B_K = [I \; 0 \; 0]^\mathsf{T}$ and $C_j = [* \; I \; 0]$. For equations (6) to be verified, it has to be $x_2 = 0$, $zx_1 = A_{11}x_1 + A_{13}x_3 - u_k$, and

$$\begin{bmatrix} 0 \\ zx_3 \end{bmatrix} = \begin{bmatrix} A_{21} & A_{23} \\ A_{31} & A_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \end{bmatrix}.$$

Case (i). Since there are no edges from the nodes K to $V \setminus \{N_j \cup K\}$, we have $A_{31} = 0$, and hence it has to be $(zI - A_{33})x_3 = 0$, i.e., z needs to be an eigenvalue of A_{33}. We now show that $x_3 \neq 0$. Suppose by contradiction that $x_3 = 0$, and that z is an invariant zero, with state-zero and input-zero direction $x = [x_1^\mathsf{T} \, 0 \, 0]^\mathsf{T}$ and $u_K = (zI - A_{11})x_1$, respectively. Then, for all complex value $\bar{z}$, the vectors x and $u_K = (\bar{z}I - A_{11})x_1$ constitute the state-zero and the input-zero direction associated with the invariant zero $\bar{z}$. Because the system is assumed to

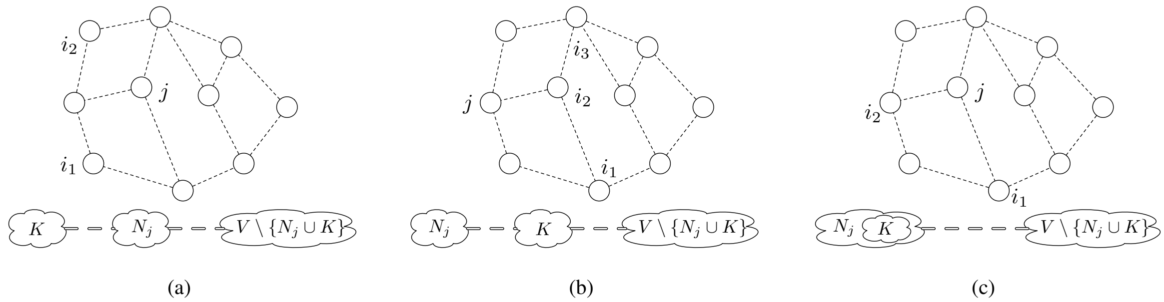


Fig. 2. The stability of the zero dynamics of a left-invertible consensus system can be asserted depending upon the location of the misbehaving agents in the network. Let j be the observer agent, and let K be the misbehaving set. Then, the zero dynamics are asymptotically stable if the set N_j separates the sets K and $V \setminus \{N_j \cup K\}$ (cfr. Fig. 2(a)), or if the set K separates the sets N_j and $V \setminus \{N_j \cup K\}$ (cfr. Fig. 2(b)), or if the set K is a subset of N_j (cfr. Fig. 2(c)).

be left-invertible, there can only be a finite number of invariant zeros [22], so that we conclude that $x_3 \neq 0$ or that the system has no zero dynamics. Because z needs to be an eigenvalue of A_{33}, and because of Lemma III.1, we conclude that the zero dynamics are asymptotically stable.

Case (ii). Since there are no edges from the nodes $V \setminus \{N_j \cup K\}$ to N_j, we have $A_{23} = 0$. We now show that $\text{Ker}(A_{21}) = 0$. Suppose by contradiction that $0 \neq x_1 \in \text{Ker}(A_{21})$. Consider the equation $(zI - A_{33})x_3 = A_{31}x_1$, and notice that, because of Lemma III.1, for all z with $|z| \geq 1$, the matrix $zI - A_{33}$ is invertible. Therefore, if $|z| \geq 1$, the vector $[(x_1)^\mathsf{T}\, 0\, ((zI - A_{33})^{-1}A_{31}x_1)^\mathsf{T}]^\mathsf{T}$ is a state-zero direction, with input-zero direction $u_K = -(zI - A_{11})x_1 + A_{13}x_3$. The system would have an infinite number of invariant zeros, being therefore not left-invertible. We conclude that $\text{Ker}(A_{21}) = 0$. Consequently, we have $x_1 = 0$ and $(zI - A_{33})x_3 = 0$, so that $|z| < 1$.

Case (iii). Reorder the variables such that the nodes N_j come before $V \setminus N_j$. For the existence of a zero dynamics, it needs to hold $x_1 = 0$ and $(zI - A_{22})x_2 = 0$. Hence, $|z| < 1$. ∎

We are left to study the case of a network with zeros outside the open unit disk, where intruders may inject non-vanishing inputs while remaining unidentified. For this situation, we only remark that a detection procedure based on an admissible region for the network state can be implemented to detect inputs evolving along unstable zero directions.

VI. Structural properties and generic solvability

In the framework of traditional control theory, the entries of the matrices describing a dynamical system are assumed to be known without uncertainties. It is often the case, however, that such entries only approximate the exact values. In order to capture this modeling uncertainty, *structured systems* have been introduced and studied, e.g., see [24], [16], [25]. Let a structure matrix $[M]$ be a matrix in which each entry is either a fixed zero or an indeterminate parameter, and let the tuple of structure matrices $([A], [B], [C], [D])$ denote the structured system

$$\begin{aligned} x(t+1) &= [A]x(t) + [B]u(t), \\ y(t) &= [C]x(t) + [D]u(t). \end{aligned} \tag{7}$$

A numerical system (A, B, C, D) is an admissible realization of $([A], [B], [C], [D])$ if it can be obtained by fixing the indeterminate entries of the structure matrices at some particular value, and two systems are structurally equivalent if they are both an admissible realization of the same structured system. Let d be the number of indeterminate entries altogether. By collecting the indeterminate parameters into a vector, an admissible realization is mapped to a point in the Euclidean space $\mathbb{R}^d$. A property which can be asserted on a dynamical system is called *structural* (or *generic*) if, informally, it holds for *almost all* admissible realizations. To be more precise, following [25], we say that a property is structural (or generic) if and only if the set of admissible realizations satisfying such property forms a dense subset of the parameters space.[8] Moreover, it can be shown that, if a property holds generically, then the set of parameters for which such property is not verified lies on an algebraic hypersurface of $\mathbb{R}^d$, i.e., it has zero Lebesgue measure in the parameter space. For instance, left-invertibility of a dynamical system is known to be a structural property with respect to the parameters space $\mathbb{R}^d$.

Let the connectivity of a structured system $([A],[B],[C])$ be the connectivity of the graph defined by its nonzero parameters. In what follows, we assume $[D]=0$, and we study the zero dynamics of a structured consensus system as a function of its connectivity. Let the generic rank of a structure matrix $[M]$ be the maximal rank over all possible numerical realizations of $[M]$.

Lemma VI.1 (Generic zero dynamics and connectivity) *Let $([A],[B],[C])$ be a k-connected structured system. If the generic rank of $[B]$ is less than k, then almost every numerical realization of $([A],[B],[C])$ has no zero dynamics.*

Proof: Since the system $([A],[B],[C])$ is k-connected and the generic rank r of $[B]$ is less than k, there are r disjoint paths from the input to the output [26]. Then, from Theorem 4.3 in [26], the system $([A],[B],[C])$ is generically left-invertible. Additionally, by using Lemma 3 in [13], it can be shown that $([A],[B],[C])$ has generically no invariant zeros. We conclude that almost every realization of $([A],[B],[C])$ has no nontrivial zero dynamics. ■

Given a structured triple $([A],[B],[C])$ with d nonzero elements, the set of parameters that make $([A],[B],[C])$ a consensus system is a subset S of $\mathbb{R}^d$, because the matrix A needs to be row stochastic and primitive. A certain property that holds generically in $\mathbb{R}^d$ needs not be valid generically with respect to the feasible set S. Let $([A],[B],[C])$ be structure matrices, and let $S\subset\mathbb{R}^d$ be the set of parameters that make $([A],[B],[C])$ a consensus system. We next show that the left-invertibility and the number of invariant zeros are generic properties with respect to the parameter space S.

Theorem VI.1 (Genericity of consensus systems) *Let $([A],[B],[C])$ be a k-connected structured system. If the generic rank of $[B]$ is less than k, then almost every consensus realization of $([A],[B],[C])$ has no zero dynamics.*

Proof: Let d be the number of nonzero entries of the structured system $([A],[B],[C])$. From Theorem VI.1 we know that, generically with respect to the parameter space $\mathbb{R}^d$, a numerical realization of $([A],[B],[C])$ has no zero dynamics. Let $S\subset\mathbb{R}^d$ be the subset of parameters that makes $([A],[B],[C])$ a consensus system. We want to show that the absence of zero dynamics is a generic property with respect to the parameter space S. Observe that S is dense in $\mathbb{R}^{d-n}$, where n is the dimension of $[A]$. Then [27], [28], it can be shown that, in order to prove that our property is generic with respect to S, it is sufficient to show that there exist some consensus systems which have no zero dynamics. To construct a consensus system with no zero dynamics consider the following procedure. Let (A,B,C) be a nonnegative and irreducible linear system with no zero dynamics, where the number of inputs is

[8] A subset $S\subseteq P\subseteq\mathbb{R}^d$ is dense in P if, for each $r\in P$ and every $\varepsilon>0$, there exists $s\in S$ such that the Euclidean distance $\|s-r\|\leq\varepsilon$.

strictly less that the connectivity of the associated graph. Notice that, following the above discussion, such system can always be found. The Perron-Frobenius Theorem for nonnegative matrices ensures the existence of a positive eigenvector x of A associated with the eigenvalue of largest magnitude r [19]. Let D be the diagonal matrix whose main diagonal equals x, then the matrix $r^{-1}D^{-1}AD$ is a consensus matrix [29]. A change of coordinates of (A, B, C) using D yields the system $(D^{-1}AD, D^{-1}B, CD)$, which has no zero dynamics. Finally, the system $(r^{-1}D^{-1}AD, D^{-1}B, CD)$ is a k-connected consensus system with, generically, no zero dynamics. Indeed, if there exists a value $\bar{z}$, a state-zero direction x_0, and an input-zero direction g for the system $(r^{-1}D^{-1}AD, D^{-1}B, CD)$, then the value $\bar{z}r$, with state direction x_0/r and input direction u, is an invariant zero of $(D^{-1}AD, D^{-1}B, CD)$, which contradicts the hypothesis. ■

Because a sufficiently connected consensus system has generically no zero dynamics, the following remarks about the robustness of a generic property should be considered. First, generic means open, i.e. some appropriately small perturbations of the matrices of the system having a generic property do not destroy this property. Second, generic implies dense, hence any consensus system which does not have a generic property can be changed into a system having this property just by arbitrarily small perturbations. We are now able to extend the result presented in [13] concerning the resilience of a linear network to external attacks to the case of linear consensus networks. In particular, we next show that a k-connected network can generically tolerate up to $\lfloor \frac{k-1}{2} \rfloor$ malicious agents or up $k-1$ faulty agents.

Theorem VI.2 (Generic identification of misbehaving agents) *Given a k-connected consensus network, generically, there exists no unidentifiable input for any set of $\lfloor \frac{k-1}{2} \rfloor$ misbehaving agents. Moreover, generically, there exists identifiable inputs for every set of $k-1$ misbehaving agents.*

Proof: Since $2\lfloor \frac{k-1}{2} \rfloor < k$, by Lemma VI.1 the consensus system with any set of $2\lfloor \frac{k-1}{2} \rfloor$ has generically no zero dynamics. By Theorem IV.3, any set of $\lfloor \frac{k-1}{2} \rfloor$ malicious agents is detectable and identifiable by every node in the network. We now consider the case of faulty agents. Let V be the set of nodes, and $K_1, K_2 \subset V$, with $|K_1| = |K_2| = k-1$, be two disjoint sets of faulty agents. Let $j \in V$. We need to show the existence of identifiable, i.e., faulty, inputs. By using a result of [26] on the generic rank of the matrix pencil of a structured system, since the given consensus network is k-connected and $|K_1| = k-1$, it can be shown that the system $(A, [B_{K_1} \ \ B_i], C_j)$, for all $i \in K_2$, is left-invertible, which confirms the existence of identifiable inputs for the current network state. By Definition 4, we conclude that the faulty set K_1 is generically identifiable by any well-behaving agent. ■

The resilience of a linear consensus network to external attacks has been characterized from a system theoretic and from a graph theoretic perspective. In the next section, we describe three algorithms to detect and identify misbehaving agents, and to ultimately recover the network functionalities from external attacks.

VII. Intrusion detection algorithms

In this section, we present three decentralized algorithms to detect and identify the misbehaving agents in a consensus network. The first procedure allows to asymptotically detect a detectable attack via local computation. The second method allows every well-behaving agent to identify the entire misbehaving set. Although these first two algorithms require only local measurements, the complete knowledge of the matrix A, and hence of the entire consensus network, is necessary for the implementation. The third algorithm, instead, requires the agents to know only a certain neighborhood of the consensus graph, and it allows for a local identification of misbehaving agents. We start by describing our first method, which is based on the following theorem.

Theorem VII.1 (Detection filter) *Let K be a set of agents and let j be a network node. Assume that the zero dynamics of the consensus system (A, B_K, C_j) are exponentially stable. Let A_{N_j} represents the N_j columns of the matrix A. The filter*

$$\begin{aligned} z(t+1) &= (A + GC_j)z(t) - GC_j x(t), \\ \tilde{x}(t) &= Lz(t) + HC_j x(t), \end{aligned} \tag{8}$$

with $G = -A_{N_j}$, $H = C_j^{\mathsf{T}}$, and $L = I - HC_j$, is such that, in the limit for $t \to \infty$, the quantity $\tilde{x}(t+1) - A\tilde{x}(t)$ is nonzero only if the input $u_K(t)$ is nonzero. Moreover, if $K \subset N_j$, then the filter (8) *asymptotically estimates the state of the network, independent of the behavior of the agents K.*

Proof: Let $G = -A_{N_j}$, and consider the estimation error $e(t+1) = z(t+1) - x(t+1) = (A + GC_j)e(t) - B_K u_K(t)$. Notice that $Le(t) = Lz(t) + C_j^{\mathsf{T}} C_j x(t) - x(t)$, and hence $\tilde{x}(t) = x(t) + Le(t)$. Consequently, $\tilde{x}(t+1) - A\tilde{x}(t) = B_K u_K(t) + Le(t+1) - ALe(t)$. By using Lemma III.1, it is a straightforward matter to show that $(A+GC_j)$ is Schur stable. If $u_K(t) = 0$, then $\tilde{x}(t+1) - A\tilde{x}(t)$ converges to zero. Suppose now that $K \subseteq N_j$. The reachable set of e, i.e., the minimum $(A + GC_j)$ invariant containing $\mathcal{B}_K$, coincides with $\mathcal{B}_K$. Indeed $(A + GC_j)\mathcal{B}_K = \emptyset$. Since $\mathcal{B}_K \subseteq \mathrm{Ker}(L)$ by construction, the vectors $Le(t)$ and $\tilde{x}(t) - x(t)$ converge to zero. ■

By means of the filter described in the above theorem, a distributed intrusion detection procedure can be designed, see [11]. Here, each well-behaving agent only implements one detection filter, making the asymptotic detection task computationally easy to be accomplished. We remark that, since the filter converges exponentially, an exponentially decaying input of appropriate size may remain undetected (see Lemma V.3 for a characterization of the effect of exponentially vanishing inputs on the final consensus value). For a finite time detection of misbehaving agents, and for the identification of misbehaving components, a more sophisticated algorithm is presented in Algorithm 1.

Theorem VII.2 (Complete identification) *Let A be a consensus matrix, let K be the set of misbehaving agents, and let c be the connectivity of the consensus network. Assume that:*

(i) *every agent knows the matrix A and $k \geq |K|$, and*

(ii) *$k < c$, if the set K is faulty, and $2k < c$ if the set K is malicious.*

Then the Complete Identification algorithm allows each well-behaving agent to generically detect and identify every misbehaving agent in finite time.

Proof: We focus on agent j. Let $k = |K|$, and let $\mathcal{K}$ be the set containing all the $\binom{n-1}{k+1}$ combinations of $k+1$ elements of $V \setminus \{j\}$. For each set $\tilde{K} \in \mathcal{K}$, consider the system $\Sigma_{\tilde{K}} = (A, B_{\tilde{K}}, C_j)$, and compute[9] a set of residual generator filters for $\Sigma_{\tilde{K}}$. If the connectivity of the communication graph is sufficiently high, then, generically, each residual function is nonzero if and only if the corresponding failure mode is active. Let K be the set of misbehaving nodes, then, whenever $K \subset \tilde{K}$, the residual function associated with the failure mode $\tilde{K} \setminus K$ becomes zero after an initial transient, so that the agent $\tilde{K} \setminus K$ is recognized as well-behaving. By exclusion, because the residuals associated with the misbehaving agents are always nonzero, the set K is identified. ■

By means of the Complete Identification algorithm, the detection and the identification of the misbehaving agents take place in finite time and independent of the misbehaving input, because the residual generators can be designed

[9] A procedure to design a residual generator filter can be found in [17].

Algorithm 1 *Complete Identification* (j-th agent)

Input : A; $k \geq |K|$;

Require : The connectivity of A to be $k+1$, if K is faulty, and $2k+1$ otherwise;

Compute the residual generators for every set of $k+1$ misbehaving agents;

while *the misbehaving agents are unidentified* **do**

- Exchange data with the neighbors;
- Update the state;
- Evaluate the residual functions;
- **if** *every i_{th} residual is nonzero* **then**
  - Agent i is recognized as misbehaving.

as dead-beat filters.[10] It should be noticed that, although no communication overhead is introduced in the consensus protocol, the Complete Identification procedure relies on strong assumptions. First, each agent needs to know the entire graph topology, and second, the number of residual generators that each node needs to design is proportional to $\binom{n-1}{k}$. Because an agent needs to update these filters after each communication round, when the cardinality of the network grows, the computational burden may overcome the capabilities of the agents, making this procedure inapplicable.

In the remaining part of this section, we present a computationally efficient procedure that only assumes partial knowledge of the consensus network but yet allows for a local identification of the misbehaving agents. Let A be a consensus matrix, and observe that it can be written as $A_d + \varepsilon\Delta$, where $\|\Delta\|_\infty = 2$, $0 \leq \varepsilon \leq 1$, and A_d is block diagonal with a consensus matrix on each of the N diagonal blocks. For instance, let $A = [a_{kj}]$, and let $V_1, \dots, V_N$ be the subsets of agents associated with the blocks. Then the matrix $A_d = [\bar{a}_{kj}]$ can be defined as

(i) $\bar{a}_{kj} = a_{kj}$ if $k \neq j$, and $k, j \in V_i$, $i \in \{1, \dots, N\}$, and

(ii) $\bar{a}_{kk} = 1 - \sum_{j \in V_i, j \neq k} a_{kj}$.

Moreover, $\Delta = 2(A - A_d)/\|(A - A_d)\|_\infty$, and $\varepsilon = \frac{1}{2}\|A - A_d\|_\infty$. Note that, if ε is "small", then the agents belonging to different groups are weakly coupled. We assume the groups of weakly coupled agents to be given, and we leave the problem of finding such partitions as the subject of future research, for which the ideas presented in [30], [31] constitute a very relevant result.

We now focus on the h-th block. Let $K = v \cup l$ be the set of misbehaving agents, where $v = V_h \cap K$, and $l = K \setminus v$. Assume that the set v is identifiable by agent $j \in V_h$ (see Section IV). Then, agent j can identify the set v by means of a set of residual generators, each one designed to decouple a different set of $|v|+1$ inputs. To be more precise, let $i \in V_h \setminus v$, and consider the system

$$\begin{bmatrix} x \\ w_v \end{bmatrix}^+ = \begin{bmatrix} A_d & 0 \\ E_v C_j & F_v \end{bmatrix} \begin{bmatrix} x \\ w_v \end{bmatrix} + \begin{bmatrix} B_v & B_i \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_i \end{bmatrix}, \quad r_v = \begin{bmatrix} H_v C_j & M_v \end{bmatrix} \begin{bmatrix} x \\ w_v \end{bmatrix}, \tag{9}$$

[10] We refer the interested reader to [17] for a design procedure of a dead beat residual generator. Notice that the possibility of detecting and identifying the misbehaving agents is, as discussed in Section IV and VI, guaranteed by the absence of zero dynamics in the consensus system.

Algorithm 2 *Local Identification* (j-th agent)

Input : A_h; $k_j \geq |K \cap V_h|$; T_h

Require : The connectivity of A_d^j to be $k_j + 1$, if K is faulty, and $2k_j + 1$ otherwise;

while *the misbehaving agents are unidentified* **do**

- Exchange data with the neighbors;
- Update the state;
- Evaluate the residual functions;
- **if** i_{th} *residual is greater than* T_h **then**
  - Agent i is recognized as misbehaving.

and the system

$$\begin{bmatrix} x \\ w_i \end{bmatrix}^+ = \begin{bmatrix} A_d & 0 \\ E_i C_j & F_i \end{bmatrix} \begin{bmatrix} x \\ w_i \end{bmatrix} + \begin{bmatrix} B_v & B_i \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_i \end{bmatrix}, \quad r_i = \begin{bmatrix} H_i C_j & M_i \end{bmatrix} \begin{bmatrix} x \\ w_i \end{bmatrix}, \tag{10}$$

where the quadruple (F_v, E_v, M_v, H_v) (resp. (F_i, E_i, M_i, H_i)) describes a filter of the form (2), and it is designed as in [17]. Then the misbehaving agents v are identifiable by agent j because v is the only set such that, for every $i \in V_h \setminus v$, it holds $r_v \not\equiv 0$ and $r_i \equiv 0$ whenever $u_v \not\equiv 0$. It should be noticed that, since A_d is block diagonal, the residual generators to identify the set v can be designed by only knowing the h-th block of A_d, and hence only a finite region of the original consensus network. By applying the residual generators to the consensus system $A_d + \varepsilon\Delta$ with misbehaving agents K we get

$$\begin{bmatrix} \hat{x} \\ \hat{w}_v \end{bmatrix}^+ = \bar{A}_{\varepsilon,v} \begin{bmatrix} \hat{x} \\ \hat{w}_v \end{bmatrix} + \begin{bmatrix} B_v & B_l & B_i \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_l \\ u_i \end{bmatrix}, \quad \hat{r}_v = \begin{bmatrix} H_v C_j & M_v \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{w}_v \end{bmatrix},$$

and

$$\begin{bmatrix} \hat{x} \\ \hat{w}_i \end{bmatrix}^+ = \bar{A}_{\varepsilon,i} \begin{bmatrix} \hat{x} \\ \hat{w}_i \end{bmatrix} + \begin{bmatrix} B_v & B_l & B_i \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_l \\ u_i \end{bmatrix}, \quad \hat{r}_i = \begin{bmatrix} H_i C_j & M_i \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{w}_i \end{bmatrix},$$

where

$$\bar{A}_{\varepsilon,v} = \begin{bmatrix} A_d + \varepsilon\Delta & 0 \\ E_v C_j & F_v \end{bmatrix}, \quad \bar{A}_{\varepsilon,i} = \begin{bmatrix} A_d + \varepsilon\Delta & 0 \\ E_i C_j & F_i \end{bmatrix}.$$

Because of the matrix Δ and the input $u_l(t)$, the residual $r_i(t)$ is generally nonzero even if $u_i \equiv 0$. However, the misbehaving agents v remain identifiable by j if for each $i \in V_h \setminus v$ we have $\|\hat{r}_v\|_\infty > \|\hat{r}_i\|_\infty$ for all $u_v \not\equiv 0$.

Theorem VII.3 (Local identification) *Let V be the set of agents, let K be the set of misbehaving agents, and let $A_d + \varepsilon\Delta$ be a consensus matrix, where A_d is block diagonal, $\|\Delta\|_\infty = 2$, and $0 \leq \varepsilon \leq 1$. Let each block h of A_d be a consensus matrix with agents $V_h \subseteq V$, and with connectivity $|K \cap V_h| + 1$. There exists $\alpha > 0$ and $u_{\max} \geq 0$, such that, if each input signal $u_i(t)$, $i \in K$, takes value in $\mathcal{U} = \{u : \varepsilon\alpha u_{\max} \leq \|u\|_\infty \leq u_{\max}\}$,[11] then each*

[11]The norm $\|u\|_\infty$ is intended in the vector sense at every instant of time. Accordingly, the misbehaving input is here assumed to be nonzero at every instant of time.

well-behaving agent $j \in V_h$ identifies in finite time the faulty agents $K \cap V_h$ by means of the Local Identification algorithm.

Proof: We focus on the agent $j \in V_h$, and, without loss of generality, we assume that $u_K(0) \neq 0$, and that the residual generators have a finite impulse response. Let $d_j = \|V_h\|$, and note that d_j time steps are sufficient for each agent $j \in V_h$ to identify the misbehaving agents. Let u^t denote the input sequence up to time t. Let $v = K \cap V_h$, $l = K \setminus v$, and observe that $\hat{r}_v(d_j) = \left[\, H_v C_j \;\; M_v \,\right] \bar{A}_{\varepsilon,v}^{d_j} \bar{x}(0) + \hat{h}_v \star u_v^{d_j - 1} + \hat{h}_l \star u_l^{d_j-1}$, where $\hat{h}_v$ and $\hat{h}_l$ denote the impulse response from u_v and u_l respectively, and $\star$ denotes the convolution operator. We now determine an upper bound for each term of $\hat{r}_v(d_j)$. Let the misbehaving inputs take value in $\mathcal{U} = \{u : \varepsilon \alpha u_{\text{max}} \leq \|u\|_\infty \leq u_{\text{max}}\}$. By using the triangle inequality on the impulse responses of the residual generator, it can be shown that $\|\hat{h}_l \star u_l^{d_j-1}\|_\infty \leq \|h_l \star u_l^{d_j-1}\|_\infty + \varepsilon c_1 u_{\text{max}} = \varepsilon c_1 u_{\text{max}}$, where h_l denotes the impulse response form u_l to r_v of the system (9), and c_1 is a finite positive constant independent of ε. Moreover, it can be shown that there exist two positive constant c_2 and c_3 such that $\|\left[\, H_v C_j \;\; M_v \,\right] \bar{A}_{\varepsilon,v}^{d_j} \bar{x}(0)\|_\infty \leq \varepsilon c_2 u_{\text{max}}$, and $\min_{u_v \in \mathcal{U}} \|\hat{h}_v \star u_v^{d_j-1}\|_\infty \geq \min_{u_v \in \mathcal{U}} \|h_v \star u_v^{d_j-1}\|_\infty - \varepsilon c_3 u_{\text{max}}$. Analogously, for the residual generator associated with the well-behaving agent i, we have $\hat{r}_i(d_j) = \left[\, H_i C_j \;\; M_i \,\right] \bar{A}_{\varepsilon,i}^{d_j} \bar{x}(0) + \hat{h}_v \star u_v^{d_j-1} + \hat{h}_l \star u_l^{d_j-1}$, and hence $\hat{r}_i(d_j) \leq \varepsilon (c_4^{(i)} + c_5^{(i)} + c_6^{(i)}) u_{\text{max}}$. Let $\bar{c} = c_1 + c_2 + c_3 + \max_{i \in V_h \setminus v} (c_4^{(i)} + c_5^{(i)} + c_6^{(i)})$, and let β be such that $\min_{u_v \in \mathcal{U}} \|h_v \star u_v^{d_j-1}\|_\infty > \beta u_{\text{min}}$. Then a correct identification of the misbehaving agents v takes place if $\beta u_{\text{min}} = \beta \varepsilon \alpha u_{\text{max}} > \varepsilon \bar{c} u_{\text{max}}$, and hence if $\alpha > \bar{c}/\beta$. ■

Notice that the constant α in Theorem VII.3 can be computed by bounding the infinity norm of the impulse response of the residual generators. An example is in Section VIII-B. A procedure to achieve local detection and identification of misbehaving agents is in Algorithm 2, where A_d^h denotes the h-th block of A_d, and T_h the corresponding threshold value. Observe that in the Local Identification procedure an agent only performs local computation, and it is assumed to have only local knowledge of the network structure.

Remark 3 It is not a trivial fact that the misbehaving agents become locally identifiable when the parameter ε is sufficiently small. Indeed, as long as $\varepsilon > 0$, the effect of the perturbation $\varepsilon \Delta$ on the residuals becomes eventually relevant with respect to the dynamics described by only A_d, preventing, after a certain time, a correct identification of the misbehaving agents [30].

VIII. EXAMPLES

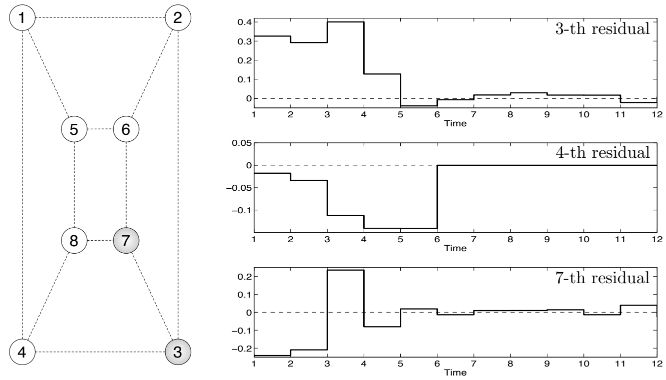
In the first example of this section the *Complete Identification* algorithm is used, while in the second example the *Local Identification* procedure is implemented.

A. Complete detection and identification

Consider the network of Fig. 3(a), and let A be a randomly chosen consensus matrix. In particular,

$$A = \begin{bmatrix} 0.2795 & 0.1628 & 0 & 0.1512 & 0.4066 & 0 & 0 & 0 \\ 0.0143 & 0.3363 & 0.3469 & 0 & 0 & 0.3025 & 0 & 0 \\ 0 & 0.0718 & 0.1904 & 0.2438 & 0 & 0 & 0.4941 & 0 \\ 0.0844 & 0 & 0.4457 & 0.0660 & 0 & 0 & 0 & 0.4040 \\ 0.1709 & 0 & 0 & 0 & 0.2694 & 0.2472 & 0 & 0.3125 \\ 0 & 0.4199 & 0 & 0 & 0.1575 & 0.3293 & 0.0932 & 0 \\ 0 & 0 & 0.0174 & 0 & 0 & 0.4241 & 0.2850 & 0.2735 \\ 0 & 0 & 0 & 0.3024 & 0.2039 & 0 & 0.2065 & 0.2873 \end{bmatrix}.$$

The network is 3-connected, and it can be verified that for any set K of 3 misbehaving agents, and for any observer node j, the triple (A, B_K, C_j) is left-invertible. Also, for any set K of cardinality 2, and for any node j, the triple (A, B_K, C_j) has no invariant zeros. As previously discussed, any well-behaving node can detect and identify up to 2



(a) (b)

Fig. 3. In Fig. 3(a) a consensus network where the nodes 3 and 7 are faulty. In Fig. 3(b) the residual functions computed by the agent 1 under the hypothesis that the misbehaving set is $\{3, 4, 7\}$.

faulty agents, or up to 1 malicious agent. Consider the observations of the agent 1, and suppose that the agents $\{3, 7\}$ inject a random signal into the network. As described in Algorithm 1, the agent 1 designs the residual generator filters and computes the residual functions for each of the $\binom{7}{3}$ possible sets of misbehaving nodes, and identify the well-behaving agents. Consider for example the system $x(t+1) = Ax(t) + B_3 u_3(t) + B_4 u_4(t) + B_7 u_7(t)$, and suppose we want to design a filter of the form (2) which is only sensible to the signal u_4. The unobservability subspace $\mathcal{S}^M_{\{3,7\}} = (\mathcal{V}^*_{\{3,7\}} + \mathcal{S}^*_{\{3,7\}})$, is

$$\mathcal{S}^M_{\{3,7\}} = \mathrm{Im}\left(\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.6624 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -0.4740 & -0.6597 & 0 \\ 0 & 0 & -0.8798 & 0.3548 & 0 \\ 0.4116 & 0 & -0.0327 & 0.0132 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0.9114 & 0 & 0.0148 & -0.0060 & 0 \end{bmatrix} \right),$$

and a possible choice for the matrices of the residual generator is

$$F = \begin{bmatrix} 0 & 0 & 0 \\ 0.0014 & -0.3222 & -0.3424 \\ -0.0013 & 0.3031 & 0.3222 \end{bmatrix}, \quad E = \begin{bmatrix} 0.2795 & 0.1628 & 0.1512 & 0.4066 \\ 0.0138 & 0.4982 & -0.2280 & 0.2003 \\ 0.0082 & -0.6095 & 0.3012 & -0.1568 \end{bmatrix},$$

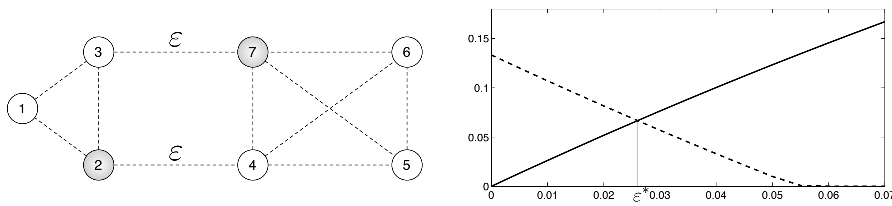
$$M = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0.9999 & 0.0128 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -0.7491 & 0.5832 & -0.3142 \end{bmatrix}.$$

It can be checked that, independent of the initial condition of the network, the residual function associated with the input 4 is zero, as in 3(b), so that the agent 4 is regarded as well-behaving. Agents 3, 7, instead, have always nonzero residual functions, and are recognized as misbehaving. If the misbehaving nodes are allowed to be malicious, then no more than 1 misbehaving node can be tolerated. Indeed, because of Theorem IV.1, there exists a set $\bar{K}$ of 4 misbehaving agents such that the system $(A, B_{\bar{K}}, C_1)$ exhibits nontrivial zero dynamics. For instance, let $\bar{K} = \{2, 4, 6, 8\}$, and note that if the initial condition $x(0)$ belongs to

$$\mathcal{V}^* = \mathrm{Im}\left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0.7842 & 0 \\ 0 & 0 & 1 \\ 0 & -0.6205 & 0 \end{bmatrix} \right),$$

then the input $u_K(t) = F_b x(t)$, where

$$F_b = \begin{bmatrix} 0 & 0 & -0.3469 & 0 & 0 & -0.1860 & 0 & 0.1472 \\ 0 & 0 & -0.4457 & 0 & 0 & 0.1966 & 0 & -0.1555 \\ 0 & 0 & 0 & 0 & 0 & -0.1063 & -0.1148 & 0.0841 \\ 0 & 0 & 0 & 0 & 0 & 0.0636 & -0.1894 & -0.0503 \end{bmatrix},$$



(a) (b)

Fig. 4. In Fig. 4(a) a consensus network with weak connections. In Fig. 4(b) the solid line corresponds to the largest magnitude of the residual associated with the well-behaving agent 3, while the dashed line denotes the smallest magnitude of the residual associated with the misbehaving agent 2, both as a function of the parameter ε. If $\varepsilon \leq \varepsilon^*$, then there exists a threshold that allows to identify the misbehaving agent 2.

is such that $y_1(t) = 0$ for all $t \geq 0$.[12] Therefore, the two systems $(A, B_{\{2,4\}}, C_1)$ and $(A, B_{\{6,8\}}, C_1)$, with initial conditions $x_1(0)$ and $x_2(0) = x_1(0) - x(0)$, and inputs

$$u_{\{2,4\}}(t) = \left[\begin{smallmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{smallmatrix}\right] F_b(x_1(t) - x_2(t)), \quad u_{\{6,8\}}(t) = \left[\begin{smallmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{smallmatrix}\right] F_b(x_2(t) - x_1(t)),$$

have exactly the same output dynamic, so that the two sets $\{2,4\}$ and $\{6,8\}$ are indistinguishable by the agent 1.

B. *Local detection and identification*

Consider the consensus network in Fig. 4(a), where $A = A_d + \varepsilon\Delta$, and

$$A_d = \begin{bmatrix} 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 0 & 1/4 & 1/4 & 1/4 & 1/4 \end{bmatrix}, \quad \Delta = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix}.$$

Let $K = \{2,7\}$ be the set of misbehaving agents, let $0.1 \leq u_2(t), u_7(t) \leq 3$ at each time t, and let $\|x(0)\|_\infty \leq 1$. Consider the agent 1, and let (F_2, E_2, M_2, H_2) and (F_3, E_3, M_3, H_3) be the residual generators as in (9) and (10) respectively, where

$$F_2 = \begin{bmatrix} -1/3 & -1/3 \\ 1/3 & 1/3 \end{bmatrix}, \quad E_2 = \begin{bmatrix} -2/3 & 0 & -1/3 \\ 2/3 & 0 & 1/3 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad H_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix},$$

and

$$F_3 = \begin{bmatrix} -1/3 & 1/3 \\ -1/3 & 1/3 \end{bmatrix}, \quad E_3 = \begin{bmatrix} -2/3 & -1/3 & 0 \\ -2/3 & -1/3 & 0 \end{bmatrix}, \quad M_3 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad H_3 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

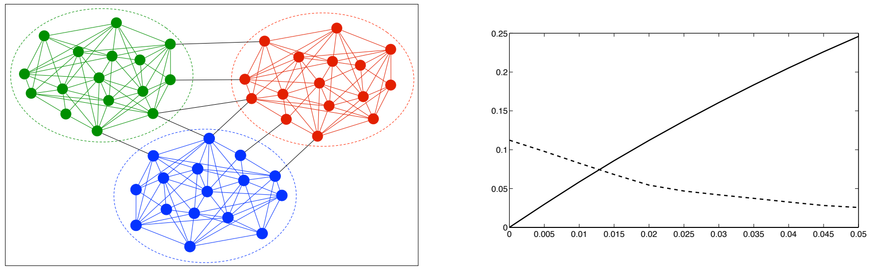
Let $\hat{h}_2^3$ (resp. $\hat{h}_7^3$) be the impulse response from the input u_2 (resp. u_7) to $\hat{r}_3$, and let u_2^1 (resp. u_7^1) denote the input signal u_2 (resp. u_7) up to time 1. Note that the misbehaving agent can be identified after 2 time steps, and that the residual associated with the agent 3 is

$$\hat{r}_3(2) = \begin{bmatrix} H_3 C_1 & M_3 \end{bmatrix} \begin{bmatrix} A_d + \varepsilon\Delta & 0 \\ E_3 C_1 & F_3 \end{bmatrix}^2 \begin{bmatrix} x(0) \\ 0 \end{bmatrix} + \hat{h}_2^3 \star u_2^1 + \hat{h}_7^3 \star u_7^1,$$

where $\star$ denotes the convolution operator. After some computation we obtain

$$\hat{r}_3(2) = \varepsilon \begin{bmatrix} H_3 C_1 & M_3 \end{bmatrix} \begin{bmatrix} A_d\Delta + \Delta A_d + \varepsilon\Delta^2 & \Delta B_2 & \Delta B_7 \\ E_3 C_1 \Delta & 0 & 0 \end{bmatrix} \begin{bmatrix} x(0) \\ u_2(0) \\ u_7(0) \end{bmatrix}$$

[12]Notice that the malicious agents need to know the entire state to implement this feedback law. The case in which the malicious agents can only implement local feedback actions is left as a direction for future research, for which the result in [12] is very relevant.



(a) (b)

Fig. 5. In Fig. 5(a) a consensus network partitioned into 3 areas. Each agent identifies the neighboring misbehaving agents by only knowing the topology of the subnetwork it belongs to. In Fig. 5(b) the smallest magnitude of the residual associated with a misbehaving agent (dashed line) and the largest magnitude of the residual associated with a well-behaving agent (solid line) are plotted as a function of ε.

and, analogously,

$$\hat{r}_2(2) = \varepsilon \left[H_2C_1 \;\; M_2 \right] \begin{bmatrix} A_d\Delta + \Delta A_d + \varepsilon\Delta^2 & \Delta B_2 & \Delta B_7 \\ E_2C_1\Delta & 0 & 0 \end{bmatrix} \begin{bmatrix} x(0) \\ u_2(0) \\ u_7(0) \end{bmatrix} + \left[H_2C_1 \;\; M_2 \right] \begin{bmatrix} A_dB_2 & B_2 \\ E_2C_1B_2 & 0 \end{bmatrix} \begin{bmatrix} u_2(0) \\ u_2(1) \end{bmatrix}$$

Recall that the agent 1 is able to identify the misbehaving agent 2 if, independent of u_2^1 and u_7^1, there exists a threshold T such that $\|\hat{r}_2(2)\|_\infty \geq T$, and $\|\hat{r}_3(2)\|_\infty < T$. The behavior of $\|\hat{r}_2(2)\|_\infty$ and $\|\hat{r}_3(2)\|_\infty$ as a function of ε is in Fig. 4(b). Note that for $\varepsilon = \varepsilon^* = 0.026$ we have $\|\hat{r}_2(2)\|_\infty = \|\hat{r}_3(2)\|_\infty = 0.07$. For instance, if $\varepsilon = 0.01$, then it can be verified that $\|\hat{r}_2(2)\|_\infty > 0.1$, and $\|\hat{r}_3(2)\|_\infty < 0.05$. It follows that a threshold $T = 0.1$ allows the agent 1 to identify the misbehaving agent 2. On the other hand, if $\varepsilon = 0.03$, then $\|\hat{r}_2(2)\|_\infty \geq 0.01$, and $\|\hat{r}_3(2)\|_\infty \leq 0.12$, so that the misbehaving agent 2 may remain unidentified. Indeed, if $x(0) = [\,1\;1\;1\;-1\;-1\;-1\;-1\,]$, $u_2^1 = u_7^1 = [\,0.1\;0.1\,]$, then $\|\hat{r}_2(2)\|_\infty = 0.01$ and $\|\hat{r}_3(2)\|_\infty = 0.12$, so that the agent 3 is recognized as misbehaving instead of the agent 2.

As a final remark, note that the larger the consensus network, the more convenient the proposed approximation procedure becomes. For instance, consider the network presented in [32], and here reported in Fig. 5(a). Such a clustered interconnection structure, in which the edges connecting different clusters have a small weight, may be preferable in many applications because much simpler and efficient protocols can be implemented within each cluster. Assume that there is a misbehaving agent in each cluster, and consider the residuals computed after 5 steps of the consensus algorithm. Let ε be the weight of the edges connecting different clusters. Fig. 5(b) shows, as a function of ε, the smallest magnitude of the residual associated with a misbehaving agent (dashed line) versus the largest magnitude of the residual associated with a well-behaving agent (solid line). If ε is sufficiently small, then our local identification method allows each well-behaving agent to promptly detect and identify the misbehaving agents belonging to the same group, and hence to restore the functionality of the network. For instance, if $\varepsilon \leq 0.01$, then, following Theorem VII.3, if the misbehaving input take value in $\{u : 0.1 \leq |u| \leq 3\}$, then a misbehaving agent is correctly detected and identified by a well-behaving agent.

IX. Conclusion

The problem of distributed reliable computation in networks with misbehaving nodes is considered, and its relationship with the fault detection and isolation problem for linear systems is discussed. The resilience of linear consensus networks to external attacks is characterized through some properties of the underlying communication graph, as well as from a system-theoretic perspective. In almost any linear consensus network, the misbehaving components can be correctly detected and identified, as long as the connectivity of the communication graph is sufficiently high. Precisely, for a linear distributed consensus network to be resilient to k concurrent faults, the connectivity of the communication graph needs to be $2k+1$, if Byzantine failures are allowed, and $k+1$, otherwise. Finally, for the faulty agents case, good performance can be obtained even when the agents do not know the entire topology of the consensus network, or when they are subject to memory or computation constraints.

Interesting aspects requiring further investigation include a quantitative study of the strong observability of a consensus network. More precisely, instead of determining whether or not a consensus network is strongly observable from the agent j, it is useful to characterize the gain between the inputs of a set of misbehaving agents and the observations of the agent j. Indeed, depending on how small such gain is, some undetectable behaviors may be neglected because not implementable by the misbehaving agents. Additionally, it is of interest to study the strong observability property under specific consensus protocols, e.g., those resulting from an optimization process on the communication weights, or those preserving the statistical properties of the initial consensus vector. Finally, the problem of clustering a large network into smaller parts is crucial for the performance of the proposed local identification procedure, and it remains the subject of future research.

References

[1] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control: Collective group behavior through local interaction," *IEEE Control Systems Magazine*, vol. 27, no. 2, pp. 71–82, 2007.

[2] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[3] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*, ser. Applied Mathematics Series. Princeton University Press, 2009, available at http://www.coordinationbook.info.

[4] N. A. Lynch, *Distributed Algorithms*. Morgan Kaufmann, 1997.

[5] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[6] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[7] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters*, vol. 53, pp. 65–78, 2004.

[8] P. Michiardi and R. Molva, "CORE: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc network," in *Communications and Multimedia Security Conference (CMS)*, Portoroz, Slovenia, Sept. 2002.

[9] L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 3, pp. 382–401, 1982.

[10] D. Dolev, "The Byzantine generals strike again," *Journal of Algorithms*, vol. 3, pp. 14–30, 1982.

[11] F. Pasqualetti, A. Bicchi, and F. Bullo, "Distributed intrusion detection for secure consensus computations," in *IEEE Conf. on Decision and Control*, New Orleans, LA, Dec. 2007, pp. 5594–5599.

[12] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterations in the presence of malicious agents - Part I: Attacking the network," in *American Control Conference*, Seattle, WA, June 2008, pp. 1350–1355.

[13] ——, "Distributed function calculation via linear iterations in the presence of malicious - Part II: Overcoming malicious behavior," in *American Control Conference*, Seattle, WA, June 2008, pp. 1356–1361.

[14] R. Patton, P. Frank, and R. Clark, *Fault Diagnosis in Dynamic Systems: Theory and Applications*. Prentice Hall, 1989.

[15] G. Basile and G. Marro, *Controlled and Conditioned Invariants in Linear System Theory*. Prentice Hall, 1991.

[16] W. M. Wonham, *Linear Multivariable Control: A Geometric Approach*, 3rd ed. Springer, 1985.

[17] M.-A. Massoumnia, G. C. Verghese, and A. S. Willsky, "Failure detection and identification," *IEEE Transactions on Automatic Control*, vol. 34, no. 3, pp. 316–321, 1989.

[18] C. D. Godsil and G. F. Royle, *Algebraic Graph Theory*, ser. Graduate Texts in Mathematics. Springer, 2001, vol. 207.

[19] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. SIAM, 2001.

[20] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1985.

[21] H. L. Trentelman, A. Stoorvogel, and M. Hautus, *Control Theory for Linear Systems*. Springer, 2001.

[22] J. Tokarzewski, *Finite Zeros in Discrete Time Control Systems*, ser. Lecture notes in control and information sciences. Springer, 2006.

[23] E. Montijano, S. Martínez, and S. Sagués, "De-ransac: Robust distributed consensus in sensor networks," *IEEE Transactions on Systems, Man & Cybernetics. Part B: Cybernetics*, May 2010, submitted.

[24] J. M. Dion, C. Commault, and J. van der Woude, "Generic properties and control of linear structured systems: a survey," *Automatica*, vol. 39, no. 7, pp. 1125–1144, 2003.

[25] K. J. Reinschke, *Multivariable Control: A Graph-Theoretic Approach*. Springer, 1988.

[26] J. van der Woude, "The generic number of invariant zeros of a structured linear system," *SIAM Journal on Control and Optimization*, vol. 38, no. 1, pp. 1–21, 1999.

[27] A. N. Kolmogorov and S. V. Fomin, *Introductory Real Analysis*. Dover Publications, 1975.

[28] K. Tchoń, "On generic properties of linear systems: An overview," *Kybernetika*, vol. 19, no. 6, pp. 467–474, 1983.

[29] H. Minc, *Nonnegative Matrices*. Wiley, 1988.

[30] R. G. Phillips and P. Kokotović, "A singular perturbation approach to modeling and control of Markov chains," *IEEE Transactions on Automatic Control*, vol. 26, no. 5, pp. 1087–1094, 1981.

[31] J. H. Chow, J. Cullum, and R. A. Willoughby, "A sparsity-based technique for identifying slow-coherent areas in large power systems," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-103, no. 3, pp. 463–473, 1984.

[32] E. Bıyık and M. Arcak, "Area aggregation and time-scale modeling for sparse nonlinear networks," *Systems & Control Letters*, vol. 57, no. 2, pp. 142–149, 2007.