49th IEEE Conference on Decision and Control
December 15-17, 2010
Hilton Atlanta Hotel, Atlanta, GA, USA

Identifying Cyber Attacks via Local Model Information

Fabio Pasqualetti, Ruggero Carli, Antonio Bicchi, and Francesco Bullo

Abstract— **This work considers the problem of detecting corrupted components in a large scale decentralized system via local model information. The electric power system, the transportation system, and generally any computer or network system are examples of large scale systems for which external (cyber) attacks have become an important threat. We consider the case of linear networks, and we model a cyber attack as an exogenous input that compromises the behavior of a set of components. We exploit two distributed methods that rely on two different sets of assumptions to achieve detection and identification. The first method takes advantage of the presence in the network of weakly interconnected subparts, it requires limited knowledge of the network model, and it affords local detection and identification of misbehaving components whose behavior deviates more than a threshold. The second method relies on the presence of a set of trustworthy leaders with better computation and communication capabilities. Only relying on a partial knowledge of the network model, the leaders cooperatively detect and identify misbehaving components.**

I. Introduction

The increasing reliance on network systems to support critical operations in defense, electric power management, and telecommunication raises the issue of reliability and robustness of such systems against external attacks. Because of the decentralized nature of network systems, cyber attacks compromising the availability of resources, the integrity of data, or the confidentiality of information are easily launched by a malignant agent. Furthermore, the growing dimension of network systems forbids any centralized implementation of an attack detection system, ruling out classical solutions as presented in [1].

The detection and the identification of misbehaving agents in a network has been the subject of intensive study among the computer scientists interested in distributed computing. In this work, we focus on the fundamental task of computing an agreement (consensus) on a variable of interest via distributed computation and in the presence of misbehaving agents. We consider the extreme case of Byzantine agents, which are omniscient, and which collude in order to cause the biggest damage to the network. In the last few years, the problem of reaching consensus in the presence of misbehaving components has been revisited from a control theoretic perspective. In these works, the network is assumed to evolve as a linear dynamical system, and the misbehaving agents are modeled as unknown and unmeasurable inputs. In [2] 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 [3], the authors provide a policy that k malicious agents can follow to prevent some of the nodes of a $2k$-connected[1] 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. Finally, in [4] the connection between the graph connectivity and the zero dynamics of a specific linear system associated with the network is explained, and a complete characterization of the policies that make a set of misbehaving agents undetectable is given. Despite the advances in the theoretical understanding of the detection and identification of misbehaving agents, efficient decentralized algorithms ensuring security against attacks are still missing. The procedures proposed so far rely indeed on an heavy combinatorial machinery to locate the attackers, they require every agent to have complete knowledge of the network structure, and they need a number of steps proportional to the cardinality of the network to converge. Therefore, although provably correct, the existing algorithms for misbehavior detection and identification are practically applicable only when the dimension of the network is relatively small.

The main contribution of this work are as follows. We present two novel methods to reduce the computational cost of the existing detection and identification algorithms. The proposed procedures rely on two different sets of assumptions, and they can be alternatively employed depending on the network model. The first method is designed to exploit the presence in a network of weakly interconnected subparts. 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. The second method admits the presence of a subset of agents with better computation and communication capabilities (leaders), and it achieves exact detection and identification even when the entire network structure is not available to any of the leaders. Under the assumption that the leaders coincide with the vertices of a connected communication graph, two algorithms

This material is based upon work supported in part by NSF grant IIS-0904501 and ARO MURI grant W911NF-05-1-0219, and in part by EC contracts IST 224428 (2008) "CHAT", and FP7-2007-2-224053 CONET.

Fabio Pasqualetti, Ruggero Carli, and Francesco Bullo are with the Center for Control, Dynamical Systems and Computation, University of California at Santa Barbara, {fabiopas,carlirug,bullo}@engineering.ucsb.edu

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

[1] The connectivity of a graph equals the maximum number of disjoint paths between any two vertices of the graph.

are proposed to distributively reconstruct the state of the network in the presence of an unknown input, and to detect the presence of a misbehaving agent. Both algorithms require only a limited knowledge of the network structure, and they are shown to converge in a finite number of steps. We conclude the paper by showing the effectiveness of our algorithms through a numerical study.

The rest of the material is organized as follows. Section II contains the problem setting. Section III describes our method to exploit the presence of weakly interconnected subnetworks, and Section IV contains an example. Sections V introduces the hierarchical structure we propose, and it contains our main results on the unknown input estimation problem and on the detection problem. Sections VI and VII contain respectively a numerical study and our conclusion.

II. Definitions and preliminary concepts

Let G denote a directed graph with vertex set $V = \{1,\dots,n\}$ and edge set $E \subseteq V \times V$. The (in)-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 contain the values x_j. A linear consensus algorithm over G is an update rule for x and it is described by the linear discrete time system

$$x(t+1) = Ax(t),$$

where the matrix A is row-stochastic and primitive, and where its (i,j)-th entry is nonzero if and only if the pair (j,i) belongs to the edge set of G. 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, $i \in V$, be the input associated with the i-th agent, and let u be the vector of the functions u_i. 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 $u_j(t) \neq 0$, and it is *well-behaving* otherwise.

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

$$x(t+1) = Ax(t) + B_K u_K(t).$$

We associate an output matrix C_j to each agent j, which describes the information about the state of the network that is directly available to j. In particular, $y_j(t) = C_j x(t)$, $C_j = [e_{n_1} \ \dots \ e_{n_p}]^T$, and $\{n_1,\dots,n_p\} = N_j$. Throughout the paper, let $\operatorname{Im}(A)$ and $\operatorname{Ker}(A)$ denote the range space and the null space defined by the matrix A.

III. Local detection and identification

In this section, after reviewing a basic filter approach to security, we characterize a topological condition that allows for local detection and identification of misbehaving agents. For ease of notation, we consider now the single misbehaving agent case. Let j be a well-behaving agent, and consider the problem of deciding whether the agent i_1 or the agent i_2 is misbehaving. Let the linear discrete time filter

$$\begin{aligned} w_{i_1}(t+1) &= F_{i_1} w_{i_1}(t) + E_{i_1} y_j(t), \\ r_{i_1}(t) &= M_{i_1} w_{i_1}(t) + H_{i_1} y_j(t), \end{aligned} \tag{1}$$

be such that $r_{i_1} \neq 0$ if and only if i_1 is misbehaving. Then, the signal r_{i_1} allows to uniquely identify the misbehaving agent i_1 against the well-behaving agent i_2. By implementing a similar filter for each possible pair[2] of misbehaving agents, the presence of the misbehaving agent i_1 is finally assessed by the agent j. A technique to design the filter (1) can be found in [4], where the knowledge of the network matrix A by the well-behaving agent j is assumed.

We consider now the case in which each well-behaving agent has a partial knowledge of the network model, and it cannot therefore design the filter presented in (1). 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 $k \neq j$, $k,j \in V_i$, $i \in \{1,\dots,N\}$, and
(ii) $\bar{a}_{kk} = 1 - \sum_{j \in V_i \setminus \{k\}} a_{kj}$, and
(iii) $\bar{a}_{kj} = 0$ otherwise.

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 the same group are strongly interacting, while the agents belonging to different groups are weakly coupled (cf. Fig. 1). We assume the groups of strongly interacting agents to be given, and we leave the problem of determining such partitions as the subject of future research, for which the ideas presented in [5] 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 $v = V_j \cap K$, and $l = K \setminus v$. Let $j \in V_h$, and consider the system (A_d, B_v, C_j). Recall from [4] that the misbehaving agents v are identifiable by agent j if the inputs u_v and u_i can be decoupled, for all $i \in V \setminus v$. To be more precise, let $^+$ denotes the time shift operation, and consider the systems

$$\begin{aligned} \begin{bmatrix} x \\ w_v \end{bmatrix}^+ &= \begin{bmatrix} A_d & 0 \\ E_v C_j & F_v \end{bmatrix} \begin{bmatrix} x \\ w_v \end{bmatrix} + \begin{bmatrix} B_v & B_i \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_i \end{bmatrix}, \\ r_v &= \begin{bmatrix} H_v C_j & M_v \end{bmatrix} \begin{bmatrix} x \\ w_v \end{bmatrix}, \end{aligned} \tag{2}$$

and[3]

$$\begin{aligned} \begin{bmatrix} x \\ w_i \end{bmatrix}^+ &= \begin{bmatrix} A_d & 0 \\ E_i C_j & F_i \end{bmatrix} \begin{bmatrix} x \\ w_i \end{bmatrix} + \begin{bmatrix} B_v & B_i \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_v \\ u_i \end{bmatrix}, \\ r_i &= \begin{bmatrix} H_i C_j & M_i \end{bmatrix} \begin{bmatrix} x \\ w_i \end{bmatrix}. \end{aligned} \tag{3}$$

[2] The design of the filter matrices depends upon the pair (i_1, i_2).

[3] The filter matrices F_v, F_i, E_v, E_i, H_v, H_i, M_v, and M_i are designed to decouple the input u_v and u_i [4].

The misbehaving agents v are identifiable by agent j if, for all $i \in V \setminus v$, we have $r_v \neq 0$ and $r_i = 0$ whenever $u_v \neq 0$. 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. Moreover, the misbehaving agents l do not affect the residuals r_i, $i \in V_h$, so that the agents v are identifiable by agent j if, for all $i \in V_h \setminus v$, we have $r_v \neq 0$ and $r_i = 0$ whenever $u_v \neq 0$. By applying the above residual generators to the consensus system $A_d + \varepsilon\Delta$ with misbehaving agents K we get

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

$$\hat{r}_v = \begin{bmatrix} H_v C_j & M_v \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{w}_v \end{bmatrix},$$

and

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

$$\hat{r}_i = \begin{bmatrix} H_i C_j & M_i \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{w}_i \end{bmatrix},$$

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, the residual r_i is generally nonzero even if $u_i = 0$. Notice that, however, the misbehaving agents v remain identifiable by j if for each $i \in V_j \setminus v$ it holds $\|\hat{r}_v\|_\infty > \|\hat{r}_i\|_\infty$ for all admissible $u_v \neq 0$.

Theorem 3.1 (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, $i \in K$, takes value in $\mathcal{U} = \{u : \varepsilon\alpha u_{\max} \leq \|u\|_\infty \leq u_{\max}\}$, then each well-behaving agent $j \in V_h$ can identify in finite time the faulty agents $K \cap V_h$.

Proof: We refer the reader to [6] for a proof of the results contained in this paper. ■

Notice that the constant α in Theorem 3.1 can be computed by bounding the infinity norm of the impulse response of the residual generators. An example follows.

IV. An Example of Local Identification

We show in this section the advantages of the clustered setup described in Section III. Consider the consensus network in Fig. 1, where $A = A_d + \varepsilon\Delta$, $\varepsilon \leq 1$, and

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

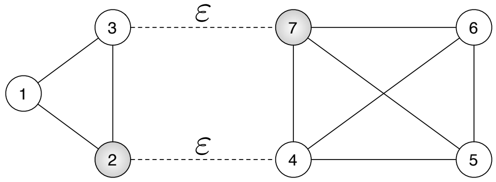

Fig. 1. A consensus network with two weakly interconnected subnetworks.

Let $K = \{2, 7\}$ be the set of misbehaving agents, 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, respectively, the residual generators as in (2) and (3), where

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

and

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

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. Because the filters (F_2, E_2, M_2, H_2) and (F_3, E_3, M_3, H_3) converge in two steps,[4] the misbehaving agent can be identified after 2 time steps. After some computation, denoting $\star$ the convolution operator, the residual associated with the agent 3 is

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

or, equivalently,

$$\hat{r}_3(1) = \varepsilon \begin{bmatrix} H_3 C_1 & M_3 \end{bmatrix} \begin{bmatrix} A_d\Delta + \Delta A_d + \varepsilon\Delta^2 & \Delta B_2 & \Delta B_7 \\ E_3 C_1 \Delta & 0 & 0 \end{bmatrix} \begin{bmatrix} x(0) \\ u_2(0) \\ u_7(0) \end{bmatrix}.$$

Analogously, we have

$$\hat{r}_2(1) = \varepsilon \begin{bmatrix} H_2 C_1 & M_2 \end{bmatrix} \begin{bmatrix} A_d\Delta + \Delta A_d + \varepsilon\Delta^2 & \Delta B_2 & \Delta B_7 \\ E_2 C_1 \Delta & 0 & 0 \end{bmatrix} \begin{bmatrix} x(0) \\ u_2(0) \\ u_7(0) \end{bmatrix}$$
$$+ \begin{bmatrix} H_2 C_1 & M_2 \end{bmatrix} \begin{bmatrix} A_d B_2 & B_2 \\ E_2 C_1 B_2 & 0 \end{bmatrix} \begin{bmatrix} u_2(0) \\ u_2(1) \end{bmatrix}.$$

The agent 1 is able to identify the misbehaving agent 2 if $\|\hat{r}_2(1)\|_\infty > \|\hat{r}_3(1)\|_\infty$ independently of u_2^1 and u_7^1. Let the inputs u_2 and u_7 take value in $\mathcal{U} = \{u : u_{\min} = \varepsilon\alpha u_{\max} \leq \|u\|_\infty \leq u_{\max}\}$. Then, it can be verified that $\|\hat{r}_2(1)\|_\infty > \|\hat{r}_3(1)\|_\infty$ if

$$\min_{u_2 \in \mathcal{U}} \left\| \begin{bmatrix} H_2 C_1 & M_2 \end{bmatrix} \begin{bmatrix} A_d B_2 & B_2 \\ E_2 C_1 B_2 & 0 \end{bmatrix} \begin{bmatrix} u_2(0) \\ u_2(1) \end{bmatrix} \right\|_\infty > 11\varepsilon u_{\max},$$

and, after some computation, if $47 < \alpha < \varepsilon^{-1}$, in which case we conclude that the agent 1 correctly identifies the misbehaving agent 2. The analysis of other possible pair of misbehaving agents can be done analogously.

[4] The eigenvalue 0 in F_2 and F_3 has algebraic (resp. geometric) multiplicity 2 (resp. 1).

V. Hierarchical estimation and detection

The previous section shows how to detect a misbehaving agent under limited knowledge of the overall system. The proposed algorithm relies on the key assumption that the magnitude of the misbehaving signal is within an interval whose size strictly depends on the parameters of the system. In this section we present an alternative method that constrains neither the input function, nor the network topology, while maintaining the assumption of local knowledge. We introduce a hierarchical structure that reduces, however, the decentralization of the network by allowing for the presence of a subset of nodes (*leaders*) with better communication and computation capabilities. Before considering the detection problem, we exploit the presence of this hierarchical structure for solving the state estimation problem in a linear system with unknown inputs. To be more precise, in Subsection V-A we propose an algorithm that allows each leader to recover the state $x(0)$ in a finite number of steps. In Subsection V-B we modify the estimation algorithm for the detection of misbehaving agents. While illustrating our algorithms we characterize also the local knowledge of the network required by each leader to accomplish the state estimation and the detection goals.

A. Hierarchical unknown input state estimation

Consider the linear network[5] $x(t+1) = Ax(t) + Bu(t)$ and let $G = (V, E)$ be the graph associated with the matrix A. Let $V^{(\ell)} = \{\ell_1, \dots, \ell_m\} \subseteq V$ denote the subset of the leaders. We assume the presence of a directed graph $G^{(\ell)} = (V^{(\ell)}, E^{(\ell)})$, where $E^{(\ell)} \subseteq V^{(\ell)} \times V^{(\ell)}$ describes the feasible communications among the leaders. We assume that $G^{(\ell)}$ is strongly connected, and we refer to it as to the *leader graph*. Let $N_i^{(\ell)}$ denote the neighbor set of the leader ℓ_i in $G^{(\ell)}$. As in Section II, the information of the state $x(t)$ directly available to the leader ℓ_i is given by $y_i(t) = C_{\ell_i} x(t)$, where C_{ℓ_i} is defined according to the neighbor set N_{ℓ_i} in G. The composite information available to the set of leaders can be conveniently described by the output matrix $C^{(\ell)} = [C_{\ell_1}^T \ \cdots \ C_{\ell_m}^T]^T$. We now show how our hierarchical setup can be conveniently used to solve the unknown input state estimation problem, in which the input matrix B is known by the leaders, while the input signal $u(t)$ is unknown and unmeasurable. For $s \in \mathbb{N}$, let

$$O_i^s = \begin{bmatrix} C_{\ell_i} \\ C_{\ell_i} A \\ C_{\ell_i} A^2 \\ \vdots \\ C_{\ell_i} A^{s-1} \end{bmatrix}, \quad Y_i^s = \begin{bmatrix} y_i(0) \\ y_i(1) \\ y_i(2) \\ \vdots \\ y_i(s-1) \end{bmatrix},$$

and

$$F_i^s = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ C_{\ell_i} B & 0 & \ddots & \ddots & 0 \\ C_{\ell_i} AB & C_{\ell_i} B & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ C_{\ell_i} A^{s-2} B & C_{\ell_i} A^{n_i-1} B & \cdots & C_{\ell_i} B & 0 \end{bmatrix}.$$

Finally, let

$$O^s = \begin{bmatrix} O_1^s \\ O_2^s \\ \vdots \\ O_m^s \end{bmatrix}, \quad Y^s = \begin{bmatrix} Y_1^s \\ Y_2^s \\ \vdots \\ Y_m^s \end{bmatrix}, \quad F^s = \begin{bmatrix} F_1^s \\ F_2^s \\ \vdots \\ F_m^s \end{bmatrix}.$$

Note that $Y^s = O^s x(0) + F^s U^s$, where U^s contains the input sequence from time 0 up to time $s-1$. From [7] we know that a system is finite-time unknown input observable (UIO), i.e., the initial state $x(0)$ can be recovered without knowing the input signal, if and only if there exists an integer $d < |V|$ such that

$$\operatorname{Ker}(O^d) = 0 \qquad \text{and} \qquad \operatorname{Im}(O^d) \cap \operatorname{Im}(F^d) = 0. \tag{4}$$

In particular conditions (4) imply that $x(0)$ can be computed as the solution of the system $Y^d = \begin{bmatrix} O^d & F^d \end{bmatrix} \begin{bmatrix} x^T & U^T \end{bmatrix}^T$. To see this, let I_n denote the n-dimensional identity matrix, and observe that

$$\operatorname{Ker} \begin{bmatrix} O^d & F^d \end{bmatrix} \perp \operatorname{Im}\left(\begin{bmatrix} I_n \\ 0 \end{bmatrix} \right) = 0, \tag{5}$$

where, given two subspaces $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A} \perp \mathcal{B}$ denotes the orthogonal projection of $\mathcal{A}$ on $\mathcal{B}$. Let

$$\begin{bmatrix} \hat{x} \\ \hat{u} \end{bmatrix} := \begin{bmatrix} O^d & F^d \end{bmatrix}^{\dagger} Y^d, \tag{6}$$

where † denotes the pseudo-inverse operation, then, from (5), we have $\hat{x} = x(0)$. Consider the basic algebraic equality

$$\operatorname{Ker}([O^d \ \ F^d]) = \cap_{i=1}^m \operatorname{Ker}([O_i^d \ \ F_i^d]), \tag{7}$$

which leads to the useful geometric interpretation of (6) that is next described. For $i \in \{1, \dots, m\}$, let $\mathcal{S}_i = \left[\begin{smallmatrix} \hat{x}_i \\ \hat{u}_i \end{smallmatrix}\right] + \mathcal{V}_i$, where $\left[\begin{smallmatrix} \hat{x}_i \\ \hat{u}_i \end{smallmatrix}\right] = [O_i^d \ \ F_i^d]^{\dagger} Y_i^d$ and $\mathcal{V}_i = \operatorname{Ker}([O_i^d \ \ F_i^d])$. Then x coincides with the projection on the subspace $\operatorname{Im}\left(\left[\begin{smallmatrix} I_n \\ 0 \end{smallmatrix}\right]\right)$ of the intersection of the affine subspaces $\{\mathcal{S}_1, \dots, \mathcal{S}_m\}$. It follows indeed from (5) and (7) that $\cap_{j=1}^m \mathcal{S}_i \perp \left[\begin{smallmatrix} I_n \\ 0 \end{smallmatrix}\right]$ results in a vector whose first n components coincide with $x(0)$. Based on the above discussion, in Algorithm 1 we propose a distributed procedure that allows each leader to compute the vector x, and that only requires local knowledge of the network. Let diam$(G^{(\ell)})$ denote the diameter of $G^{(\ell)}$.

Theorem 5.1 (Convergence of Algorithm 1): Let $(A, B, C^{(\ell)})$ be the unknown input linear system associated with the graph G and the leader graph $G^{(\ell)}$. Assume that

(i) $G^{(\ell)}$ is strongly connected, and
(ii) there exists an integer d such that $\operatorname{Ker}(O^d) = 0$ and $\operatorname{Im}(O^d) \cap \operatorname{Im}(F^d) = 0$,[6] and

[5] The results presented in this section hold for general linear networks, i.e., they are not restricted to consensus dynamics.

[6] This condition ensures the solvability of the unknown input state estimation problem [7].

Algorithm 1: *Decentralized state estimation (leader i)*

```
Input    : $O_i^d$, $Y_i^d$, $F_i^d$;
Require  : $\mathrm{Ker}(O^d) = 0, \mathrm{Im}(O^d) \cap \mathrm{Im}(F^d) = 0$;
set $[\hat{x}_i^T \ \hat{u}_i^T]^T = ([O_i^d \ \ F_i^d])^\dagger Y_i^d$, $\mathcal{V}_i = \mathrm{Ker}([O_i^d \ \ F_i^d])$;
transmit $\mathcal{S}_i = [\hat{x}_i^T \ \hat{u}_i^T]^T + \mathcal{V}_i$;
while $\mathcal{V}_i \perp \mathrm{Im}([I_n \ 0]^T) \neq 0$ do
    for $\ell_j \in N_i^{(\ell)}$ do
        receive $[\hat{x}_j^T \ \hat{u}_j^T]^T$ and $\mathcal{V}_j$;
        set $[\hat{x}_i^T \ \hat{u}_i^T]^T = [\hat{x}_i^T \ \hat{u}_i^T]^T \perp (\mathcal{S}_i \cap \mathcal{S})_|$,
        $\mathcal{V}_i = \mathcal{V}_i \cap \mathcal{V}_j$;
    transmit $\mathcal{S}_i = [\hat{x}_i^T \ \hat{u}_i^T]^T + \mathcal{V}_i$;
return $\hat{x}_i$;
```

Algorithm 2: *Decentralized detection (leader i)*

```
Input    : $O_i^d$, $Y_i^d$;
Require  : $\mathrm{Im}(O^d) \cap \mathrm{Im}(F^d) = 0$;
set $\mathcal{S}_i = (O_i^d)^\dagger Y_i^d + \mathrm{Ker}(O_i^d)$;
transmit $\mathcal{S}_i$;
for diam$(G^{(\ell)})$ iterations do
    for $j \in N_i^{(\ell)}$ do
        receive $\mathcal{S}_j$;
        set $\mathcal{S}_i = \mathcal{S}_i \cap \mathcal{S}_j$;
    transmit $\mathcal{S}_i$;
if $\mathcal{S}_i = \emptyset$ then return 1
else return 0
```

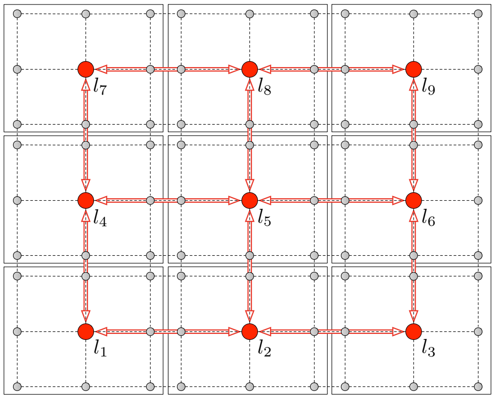

Fig. 2. A grid partitioned into 9 cblocks. Each block is identical and it contains 9 vertices. The central vertex of a block corresponds to the leader.

(iii) each leader i knows the matrices O_i^d and F_i^d.

The *Decentralized state estimation* algorithm provides each leader with the system initial state in diam$(G^{(\ell)})$ steps.

Remark 1 (Network knowledge): The computation of the matrices O_i^d, $i \in V^{(\ell)}$, does not require the knowledge of the entire network model. Given a graph, let a path be a sequence of vertices, such that any two consecutive vertices in the sequence are connected through an edge. Let the length of a path equal the number of its edges. Let A be the network matrix, and observe that the (i, j)-th entry of A^k, with $k \in \mathbb{N}$, is nonzero if and only if there exists a path of length k connecting the agent j to i. Let $N_{\ell_i}^d \subseteq V$ denote the set of agents connected to ℓ_i through a path of length at most d. It can be shown that the matrix O_i^d can be computed by only knowing the sub-matrix of A with rows and columns in $N_{\ell_i}^d$, and hence only a subnetwork of the consensus network.

We conclude this section with a remark on the convergence of Algorithm 1. The Decentralized state estimation algorithm may converge before diam$(G^{(\ell)})$ iterations. For instance, if the pair (A, C_{ℓ_1}) is observable, then, with a sufficiently large number of observations d, the leader ℓ_1 is able to reconstruct the state without communicating with the other leaders. Note however that a larger d requires the leaders to know a larger subnetwork, and, as it is shown in Section VI, it introduces numerical difficulties in the execution of our algorithm.

B. Hierarchical detection

We consider now the problem of detecting the presence of misbehaving agents. Because the misbehaving set is a priory unknown, the input matrix B and hence the matrix F^d are to be considered unknown as well. Let $U^d = [u(0)^T \ \ldots \ u(d-1)^T]^T$, and assume that each leader ℓ_i has collected the observations $y_{\ell_i}(0), \ldots, y_{\ell_i}(d-1)$. In Algorithm 2 we propose a procedure that allows the leaders to detect if $F^d U^d \neq 0$ without using the matrix F^d.

Theorem 5.2 (Convergence of Algorithm 2): Let $(A, B, C^{(\ell)})$ be the unknown input linear system associated with the graph G and the leader graph $G^{(\ell)}$. Let u be the misbehaving input, and let $U^d = [u(0)^T \ \cdots \ u(d-1)^T]^T$, with $d \in \mathbb{N}$. Assume that

(i) $G^{(\ell)}$ is strongly connected, and

(ii) $\mathrm{Im}(O^d) \cap \mathrm{Im}(F^d) = 0$,[7] and

(iii) each leader i knows the matrices O_i^d.

Then the *Decentralized detection* algorithm allows each leader to detect if $F^d U^d \neq 0$ in at most diam$(G^{(\ell)})$ steps.

The following remarks are in order. First, the condition $\mathrm{Ker}(O^d) = 0$ is not required by the Decentralized detection because only the presence of an unknown input has to be assessed. Second, in order to detect a misbehaving input that becomes nonzero at an arbitrary instant of time, the detection algorithm needs to be executed iteratively. Precisely, at each time $t \geq d-1$, the consistency of the system $Y_t^d = O^d x(t-d+1)$ needs to be checked with the detection algorithm, where $Y_t^d = [y(t-d+1)^T \ \cdots \ y(t)^T]^T$, $y(t) = C^{(\ell)} x(t)$, and $U_t^d = [u(t-d+1)^T \ \cdots \ u(t)^T]^T$. Third and finally, for the detection to be possible, there must exist $d \in \mathbb{Z}$ and $t \geq d-1$ such that $F^d U_t^d \neq 0$. Such condition coincides with the left-invertibility of the linear network, which has to be assumed by any detection method [7].

VI. An example of hierarchical estimation and detection

We show in this section the main advantages over classical solutions of the hierarchical structure presented in Section V.

[7] This condition ensures the detectability of the misbehaving input by means of d measurements [7].

Let the network G be a two dimensional lattice with $(ab)^2$ agents, and let the network be partitioned into b^2 identical blocks containing a^2 vertices each. An example with $b = 3$ and $a = 3$ is in Fig. 2. Let A describe a linear consensus algorithm on G. Let V_i, with $i \in \{1, \dots, b^2\}$, denote the set of agents belonging to the i-th block, and let the central vertex $l_i \in V_i$ represent the i-th leader. We assume that the leaders l_i and l_j are connected through an undirected edge if there exists $h_1 \in V_i$ and $h_2 \in V_j$ that are connected in G. We focus on leader l_1 and we analyze the performance of Algorithm 1 and 2 as a function of the parameters a and b.

A. State estimation

We compare here the performance of Algorithm 1 with the method proposed in [8], where the state is recovered by relying on the observability property of the pair (A, C_{l_1}). We show that, although theoretically correct, the latter method suffers from numerical instability when the dimension of A grows. Let $a = 3$ and $b = 3$, and compute the condition number[8] of the observability matrix of the pair $(A, C_k^{(\ell)})$, where $C_k^{(\ell)}$ is the composite output matrix associated with the leader set $V_k^{(\ell)} = \{l_1, \dots, l_k\}$, $k = 1, \dots, 9$. As we see from Table I, the condition number rapidly decreases by increasing the number of leaders. To be more precise in the case of $V_1^{(\ell)}$ the condition number of the observability matrix results to be $\sim 10^{14}$ so that the problem of estimating the state only relying on the measurements of l_1 is very ill-conditioned. When the leader set is $V_9^{(\ell)}$, the condition number becomes $\sim 10^2$, so that each leader can reliably estimate the correct state via distributed computation by means of Algorithm 1. We now let $|V^{(\ell)}| = b^2$ and we show a scalability property of Algorithm 5.1. Let d be the smallest number of measurements that ensures the full rank of the observability matrix, i.e., such that $\text{Ker}(O^d) = 0$. Observe from Table I that, when a is fixed and b grows, the number d remains constant. We conclude that, for the particular network structure considered in this section, the part of the network that a leader needs to know to estimate the network state does not depend upon the cardinality of the network.

TABLE I

Leader	Condition number	Size (a)	Size (b)	Measurement (d)
$V_1^{(\ell)}$	$\sim 10^{14}$	3	1	2
$V_2^{(\ell)}$	$\sim 10^{7}$	3	3	3
$V_3^{(\ell)}$	$\sim 10^{5}$	3	5	3
$V_4^{(\ell)}$	$\sim 10^{4}$	3	7	3
$V_5^{(\ell)}$	$\sim 10^{4}$	5	1	6
$V_6^{(\ell)}$	$\sim 10^{3}$	5	3	7
$V_7^{(\ell)}$	$\sim 10^{3}$	5	5	7
$V_8^{(\ell)}$	$\sim 10^{2}$	5	7	7
$V_9^{(\ell)}$	$\sim 10^{2}$	5	9	7

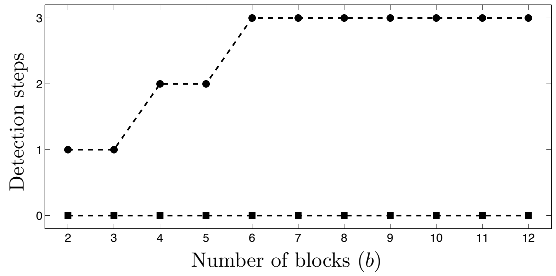

Fig. 3. The figure shows the number of iterations required for the detection of a misbehaving agent by means of Algorithm 2. Both the cases of $a = 3$ (squares) and $a = 5$ (circles) are plotted as a function of b.

B. Detection

We analyze here the performance of Algorithm 2. Let the agent $i \in V \backslash V^{(\ell)}$ be misbehaving, and let the input sequence $\{u_i(t),\ t \in \mathbb{Z}_{\geq 0}\}$ be an i.i.d. sequence taking value in the interval $(0, 1)$. For each $a \in \{3, 5\}$ and each $b \in \{2, \dots, 12\}$ we consider 20 randomly chosen consensus weights, we locate b^2 leaders (cfr. Fig. 2), and we choose the misbehaving agent i. The first instant of time at which a leader detects the presence of i by means of Algorithm 2 is reported in Fig. 3. Note that the detection time remains constant when the dimension of the network grows beyond a threshold. Hence, Algorithm 2 converges before $\text{diam}(G^{(\ell)})$ iterations, and it exhibits therefore desirable scalability properties.

VII. Conclusion

The problem of estimating the state of a linear network and the problem of detecting misbehaving parts in a linear network have been considered. Whereas classical approaches require a complete knowledge of network model, our methods only assume partial knowledge of the system structure.

References

[1] S. X. Ding, *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer, 2008.

[2] 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.

[3] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterations in the presence of malicious - Part II: Overcoming malicious behavior," in *American Control Conference*, Seattle, WA, Jun. 2008, pp. 1356–1361.

[4] F. Pasqualetti, A. Bicchi, and F. Bullo, "Consensus computation in unreliable networks: A system theoretic approach," *IEEE Transactions on Automatic Control*, Feb. 2010, submitted.

[5] 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.

[6] F. Pasqualetti, R. Carli, A. Bicchi, and F. Bullo, "Identifying cyber attacks under local model information - Proof of results," 2010, available at http://fabiopas.it/papers/FP-RC-AB-FB-10f-bis.pdf.

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

[8] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation and consensus using linear iterative strategies," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 650–660, 2008.

[8] The condition number equals the ratio of the largest singular value to the smallest. Large condition numbers indicate a nearly singular matrix. Here, the pair $(A, C_k^{(\ell)})$ is assumed to be observable for each $k \in \{1, \dots, 9\}$.