Secure State-Reconstruction Over Networks Subject to Attacks

Yanwen Mao, *Graduate Student Member, IEEE*, Suhas Diggavi, *Fellow, IEEE*, Christina Fragouli, and Paulo Tabuada, *Fellow, IEEE*

***Abstract*—Secure state-reconstruction is the problem of reconstructing the state of a linear time-invariant system from sensor measurements that have been corrupted by an adversary. Whereas most work focuses on attacks on sensors, we consider the more challenging case where attacks occur on sensors as well as on nodes and links of a network that transports sensor measurements to a receiver. In this letter we provide necessary and sufficient conditions for the secure state-reconstruction problem to be solvable in the presence of attacks on sensors and on the network.**

***Index Terms*—Fault tolerant systems, network analysis and control, fault detection.**

I. Introduction

THIS letter is concerned with the problem of reconstructing the state of a linear time-invariant system from measurements that have been corrupted by an adversary. Several examples of such attacks have been reported in [1], [2]. Moreover, the increasingly distributed and interconnected nature of Cyber-Physical Systems (CPS), including IoT devices, creates new opportunities for such attacks [3], [4]. Hence the security of CPSs is a problem of vital importance [5], [6].

The problem studied in this letter, state reconstruction despite attacks on the information to be processed, is termed the Secure State-Reconstruction (SSR) problem [7], which we will discuss in more detail in Section III-E. Most literature on the SSR problem has focused on determining necessary and sufficient conditions on the number of attacked sensors for solvability [8], [9], on efficient algorithms [10]–[17], and more recently on the algorithmic complexity of this problem [7].

Although none of the above papers considers networks, it is known that communication channels in CPS are vulnerable to certain types of attacks [4]. A related line of work, [18], [19], considers the problem of distributed estimation with sensor measurements subject to attacks and, in particular, [18] presents necessary conditions for this problem to be solvable. We note that the SSR problem becomes much more intricate when the adversary attacks not only the sensors but also the network that transports the sensor measurements to the location where they are processed. In this letter we go beyond the results proposed in [18] by providing necessary and sufficient conditions for the SSR problem to be solved in the more general setting where both sensors and the communication network are under attack. A different line of work addresses the resiliency of communication networks while being agnostic to what the transmitted data will be used for, the most prominent line of work is secure network coding [20] where the amount of secure information rate can be expressed in term of min-cut values. For example, [21] develops achievability protocols and outer bounds for the secure network coding problem where the edges are subject to packet erasures, and [22] explores the capacity region of a quadratically constrained channel corrupted by a causal adversary. Another example comes from the use of error-correction coding techniques, see [23], [24], which can be used to detect attacked links and even recover information from partially corrupted messages.

In contrast with the previously cited literature that mostly focused on attacks only on sensors or attacks only on network links and nodes, in this letter we consider the scenario where sensor measurements are subject to attacks and transported to a receiver location by a network whose nodes and links are also subject to attacks. In this context we ask: how many sensors, network links, and network nodes can be attacked while ensuring solvability of the SSR problem? We give a complete solution to this question by providing necessary and sufficient conditions for solvability of the SSR problem.

The rest of this letter is organized as follows. In Section II, we define the notation used throughout this letter. In Section III, we introduce the system model, the network model and give a formal definition of the SSR problem. As our main result, we introduce necessary and sufficient conditions for the SSR problem to be solvable in Section IV. The sufficiency of these conditions is proved in Section V, by designing a

Manuscript received March 15, 2020; revised May 15, 2020; accepted June 3, 2020. Date of publication June 9, 2020; date of current version June 23, 2020. This work was supported in part by the Army Research Laboratory under Cooperative Agreement under Grant W911NF-17-2-0196, in part by UC-NL under Grant LFR-18-548554, and in part by NSF under Award 1740047 and Award 1705135. Recommended by Senior Editor F. Dabbene. *(Corresponding author: Yanwen Mao.)*

The authors are with the Department of Electrical and Computer Engineering, University of California at Los Angeles, Los Angeles, CA 90095 USA (e-mail: yanwen.mao@ucla.edu; suhas@ucla.edu; christina.fragouli@ucla.edu; tabuada@ucla.edu).

Digital Object Identifier 10.1109/LCSYS.2020.3000853

2475-1456 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: UCLA Library. Downloaded on January 13,2022 at 02:57:08 UTC from IEEE Xplore. Restrictions apply.

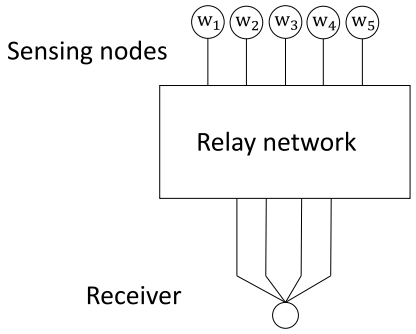

Fig. 1. Measurements are transmitted to the receiver via a relay network.

coding algorithm, whereas necessity is proved in Section VI by proposing an attacking strategy for the adversary/attacker.

II. Notation

We denote a directed acyclic graph by $G = (V, E)$ where V is the set of vertices and $E \subseteq V \times V$ is the set of edges. An edge from vertex i to j is denoted by (i, j). For an edge $(i, j) \in E$, we call vertex i an in-neighbor of vertex j, and vertex j an out-neighbor of vertex i. By removing a vertex k from the graph, we remove vertex k as well as all edges coming into or out of it.

The cardinality of a finite set $I = \{i_1, \ldots, i_p\}$ is denoted by $|I| = p$. For matrices $Q_{i_1}, Q_{i_2}, \ldots, Q_{i_p}$ over the same field and with the same number of columns, the matrix Q_I is defined by stacking $Q_{i_1}, Q_{i_2}, \ldots, Q_{i_p}$ vertically, in other words, $Q_I = \left[Q_{i_1}^T | Q_{i_2}^T | \cdots | Q_{i_p}^T\right]^T$.

III. Problem Setting

A. System Model

Consider a linear time-invariant dynamical system:

$$x[k+1] = Ax[k], \tag{1}$$

where $k \in \mathbb{N}$ is the time index, $x[k] \in \mathbb{R}^n$ is the state vector and $A \in \mathbb{R}^{n \times n}$ is the system matrix. The system is monitored by a set P of p sensors $\{w_1, w_2, \ldots, w_p\}$, and the ith sensor w_i measures the state according to:

$$y_i[k] = C_i x[k], \quad i = 1, 2, \ldots, p, \tag{2}$$

where $y_i[k] \in \mathbb{R}^{\nu_i}$ and $C_i \in \mathbb{R}^{\nu_i \times n}$. In the context of this letter, the specific values of the observability indices do not play a role, hence we assume n to be the observability index for all (A, C_i) without loss of generality. By collecting n measurements, we can write the observations from the ith sensor as follows:

$$Y_i[k] = \mathcal{O}_i x[k], \tag{3}$$

where $Y_i[k] = \left[y_i^T[k] | y_i^T[k+1] | \ldots | y_i^T[k+n-1]\right]^T$ and $\mathcal{O}_i$ is the observability matrix of sensor i, which is defined by $\mathcal{O}_i = \left[C_i^T | (C_i A)^T | \ldots | (C_i A^{n-1})^T)\right]^T$.

B. Network Model

As shown in Fig. 1, sensing nodes, i.e., network nodes equipped with sensors, transmit their measurements via a synchronous relay network to the receiver. We model the relay network, together with the sensing nodes, by a directed graph $G = (P \cup V, E)$, where P and V represent the set of sensing nodes and relay nodes (nodes in the relay network) respectively, and $(i, j) \in E$ represents the direct communication link from $i \in P \cup V$ to $j \in P \cup V$. Note that a node can both be a sensing node and a relay node, i.e., it is possible that $P \cap V \neq \varnothing$.

We assume the following properties about the network.

1) The pair (A, C_P) is observable.[1]
2) Each node has a unique identifier, and knows the identity of the in-neighbor from which it receives a message.
3) Each communication link has infinite capacity, i.e., we assume that real numbers can be transmitted through a link. Moreover, an attack-free link reliably transmits messages, i.e., messages are not lost, delayed, neither corrupted.
4) The set of attacked links and nodes does not change over time.
5) The network operates synchronously. In other words, all the nodes receive, compute, and transmit messages in a synchronized manner and this process takes exactly one time step.

Among all these assumptions, the system observability (Assumption 1) and stability of the attack (Assumption 4) are standard in SSR problem formulations, see [8], [12], [25]. The unique identifier and local knowledge of the network (Assumption 2) are also widely used, see [18], [26]. Moreover, the synchronous communication network model (Assumption 5) is widely accepted, see [27]. Although we make Assumption 3 to avoid working over finite fields, we note that our impossibility results for networks with infinite capacity directly carry over to networks with finite capacity. Also note that the network can be either wired, in which case nodes can send different messages to different out-neighbors at the same time instant; or wireless, where all out-neighbors of a node receive the same message from this node at the same time. More detailed network models, including asynchrony and links with capacities, make the problem more challenging and are left to future work.

C. Adversary Model

We consider that a subset $M_1 \subseteq E$ of links and a subset $M_2 \subseteq (P \cup V)$ of nodes is subject to active attacks. We assume the adversary is omniscient, i.e., it is aware of the system state, the measurements of all sensing nodes, the network topology, etc. In terms of capability, an attacked link is allowed to arbitrarily alter messages passing through it, including erasing, dropping or delaying the messages, and an attacked node may arbitrarily deviate from any prescribed rules and thus send any message to its out-neighbors. Attacked nodes are also

[1]Recall that C_P is the matrix obtained from $C = \left[C_1^T | C_2^T | \ldots | C_p^T\right]^T$ by removing all the rows whose indices are not in P.

Authorized licensed use limited to: UCLA Library. Downloaded on January 13,2022 at 02:57:08 UTC from IEEE Xplore. Restrictions apply.

allowed to send different messages to different out-neighbors at the same time instant under the wired network assumption. Moreover, attacked links and nodes are allowed to work cooperatively. The only assumption we make is that the power of the adversary is limited, i.e., the number of attacked links and nodes is upper bounded by f_1 and f_2. In other words, the following inequalities hold $|M_1| \leq f_1$ and $|M_2| \leq f_2$. Note that f_1 and f_2 are known to the receiver.

D. Definitions

Intuitively, in order for the receiver to reconstruct the state despite the existence of attacked nodes and links, the network must possess 2 kinds of redundancies: measurement redundancy and transmission redundancies, which can be formulated through the notions of critical set (first introduced in [18]) and mix cut with respect to (w.r.t.) a critical set.

Definition 1 (Critical Set): A set $S \subseteq P$ is said to be a critical set if the pair $(A, C_{P\setminus S})$ is not observable.

Definition 2 (Mix Cut With Respect to a Critical Set): Consider a critical set S. A set $H_S \subseteq (P \cup V \cup E)$ is called a mix cut w.r.t. S if removal of $H_S \cap (V \cup E)$ from G disconnects the receiver and $H_S \cap P$. We denote by $L(H_S) = |H_S \cap E|$ the number of links in the mix cut and $N(H_S) = |H_S \cap (P \cup V)|$ the number of nodes in the mix cut.

E. The Secure State-Reconstruction Problem

We now formally define the secure state-reconstruction problem studied in this letter.

Problem 1 (Secure State-Reconstruction (SSR) Problem):

Input: a linear system, defined by (1) and (2), and satisfying Assumption 1, a network satisfying Assumptions 2, 3, and 4, the numbers f_1 of attacked links and f_2 of attacked nodes.

Question: Is it possible for the receiver to reconstruct the initial value $x[0]$ of the state of the linear system, provided as input, from received messages, knowing that at most f_1 links and f_2 nodes in the network are subject to the actions of an attacker respecting assumption 5?

Note that in our problem setting, the receiver has no knowledge about the network topology. Although the formal statement of the SSR problem only requires the initial value $x[0]$ of the state to be reconstructed, all the results in this letter can be suitably extended to the reconstruction of the value $x[t]$ of the state at any time $t \in \mathbb{R}_0^+$.

IV. Main Result

We present the main contribution of this letter in the next result. Note that it can be conveniently extended to the case of multiple receivers (i.e., MIMO) who want to reconstruct the same initial value $x[0]$ of the state.

Theorem 1: The SSR problem is solvable if and only if for every critical set S and every mix cut H_S w.r.t S, the following bounds are satisfied:

$$L(H_S) > 2f_1 \text{ or } N(H_S) > 2f_2.$$

Theorem (1) allows us to recover several results in the literature. To do so, however, we need to introduce the notion of minimum cut.

Definition 3 (Minimum Cut): A cut between a critical set S and the receiver is a set of links by removing which disconnects S from the receiver. A minimum cut L_S between S and the receiver is a cut with the smallest cardinality (if there are multiple, any of them is a minimum cut).

If the attacked units are restricted to: (1) communication links, or (2) sensing nodes and relay nodes, we obtain the following corollaries.

Corollary 1: If there are no attacked nodes, i.e., $f_2 = 0$, the SSR problem is solvable if and only if for every critical set S, $f_1 < |L_S|/2$.

This corollary states that if the adversary is only able to attack communication links, then the SSR problem is solvable if and only if strictly less than half of the links in the minimum cut between any critical set S and the receiver are attacked. Similar results are known in the network coding [20], although the setting is slightly different due to finite capacity constraints.

Corollary 2: If there are only attacked nodes, i.e., $f_1 = 0$, the SSR problem is solvable if and only if removal of any subset of $2f_2$ nodes does not disconnect any critical set S and the receiver.

Note that the "only if" part of this corollary has been proved in [18, Th. 1]. Similarly to corollary (1), if there are only attacked nodes, then the SSR problem is solvable if and only if strictly less than half of the nodes in any cut between any critical set S and the receiver are attacked.

A preliminary version of this result when each node has one attack-free communication link to the receiver (i.e., $f_1 = 0$) was first proposed in [8]. This simpler version of the SSR problem is proved to be solvable if and only if the pair (A, C) remaining observable even after the removal of any $2f_2$ sensing nodes. This is also known as $2f_2-$sparse observability, which was first proposed implicitly in [25] and then explicitly in [12], [13].

V. Sufficiency: Coding and Decoding Algorithms

To show that the SSR problem is solvable when the sufficient condition is met, we design a coding algorithm and prove that the receiver is able to reconstruct the initial value $x[0]$ of the state.

A. Encoding Algorithm

We ask each node in the relay network to send the same message to all its out-neighbors. Each message is composed of several bulletins. Each bulletin has 2 sections: the information section and the routing section.

To start a transmission, each sensing node creates a bulletin with its measurement in the information section, and leaves the routing section blank. Whenever a message arrives from an in-neighbor, a relay node attaches the identifier of the in-neighbor at the end of the routing section of each bulletin of the received message, while keeping the information section untouched.

After receiving and processing messages from all its in-neighbors, the relay node creates a new message whose measurement section consists of the concatenation of the bulletins in the measurements sections of the processed packages.

Authorized licensed use limited to: UCLA Library. Downloaded on January 13,2022 at 02:57:08 UTC from IEEE Xplore. Restrictions apply.

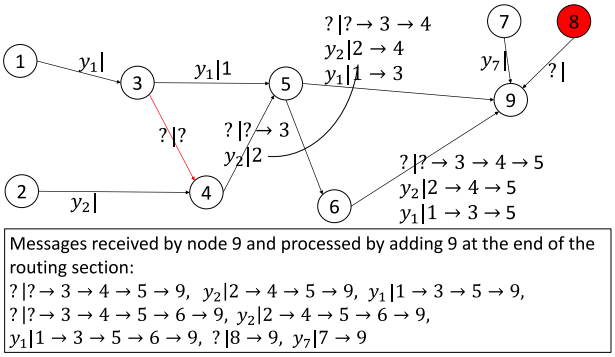

Fig. 2. A simple illustration of the encoding algorithm. Nodes 1, 2, 7, and 8 are sensing nodes, 3, 4, 5, and 6 are relay nodes, and node 9 is the receiver. Node 8 and the link from node 3 to node 4 are attacked and denoted in red. As an illustration, node 6 sends to node 9 a message composed of 3 bulletins. Each bulletin has an information section and a routing section, divided by a vertical line. For example, in the third bulletin, $y_1|1\to 3\to 5$, y_1 is the value of the information section and $1\to 3\to 5$ is the value of the routing section.

The sequence by which these bulletins are concatenated will be of no consequence for the algorithm's correctness. The relay node then sends this message to all its out-neighbors.

At last, the receiver attaches the identifier of the in-neighbor followed by its own identifier at the end of the routing section of each bulletin whenever it receives a message. For a bulletin that does not pass through any attacked link or node, the sequence of node identifiers in its routing section shows the route it has passed from the sensing node to the receiver. Therefore we regard two consecutive node identifiers in the routing section as a link.

Fig. 2 provides a simple illustration of our encoding algorithm. Since attacked links and nodes, which are colored in red, can arbitrarily change messages passing through, we denote by "?" the messages transmitted by them. All messages are represented on the figure. Note that the adversary should still keep the format of the message (i.e., bulletins, an information section and a routing section), otherwise the receiver or a relay node can easily detect it has been corrupted by an adversary and erase it.

The following Lemma is a direct consequence of the above described protocol.

Lemma 1: If a message sent by a sensing node does not pass through any attacked link or attacked node, the receiver will receive a bulletin with the sensing node's measurement in the information section and the routing in the routing section.

We will use this lemma to show that if a bulletin is relayed by attacked links or nodes, then at least one of them is recorded in its routing section.

Lemma 2: For each bulletin in each received message by the receiver, if the first identifier i in the routing section does not belong to a sensing node, or the information section value $\tilde{y}_i[0]$ does not satisfy $\tilde{y}_i[0] = C_i x[0]$, then there is at least 1 attacked link or node in the routing section.

Proof: By Lemma (1) we know that if a bulletin from node w_i does not pass through attacked links and nodes, the value in the information section of the corresponding bulletin satisfies $\tilde{y}_i[0] = C_i x[0]$. In other words, if $\tilde{y}_i[0] \neq C_i x[0]$ then this bulletin must have been altered and relayed by some attacked links or nodes. ■

Now we show that the last attacked link or node this bulletin passes must be recorded in the routing section of the bulletin. Consider the first non-attacked node (note that it can also be the receiver) this bulletin passes through after being altered by the last attacked link or node in its route. By assumption, the non-attacked node knows the identifier of its in-neighbors, this node will attach the in-neighbor's identifier in the routing section, and thereby automatically adds the link from the in-neighbor to itself, with at least one of them being attacked. And since all upcoming nodes and links are non-attacked, the identifier of the attacked link or node will be transmitted to the receiver in the information section of the bulletin.

B. Decoding Algorithm

We first notice that for every bulletin with correct information, the receiver can determine the time at which the message was sent by counting the number of links the message transversed.

The receiver is able to reconstruct the initial value $x[0]$ of the state after it receives all bulletins generated by non-attacked sensing nodes at time instances $0, 1, \ldots, n-1$. To do this, the receiver analyzes each bulletin in each message and stores those sent between 0 and $n-1$ (both included) until it is convinced that all bulletins generated during this period and relayed by only non-attacked links and nodes are received.[2]

The naive decoding algorithm is as follows:

Step 1: The receiver categorizes all stored bulletins into different classes according to the information in their routing sections: each different path corresponds to a different class.

Step 2: For each class j, the receiver computes the time instance when each bulletin in it was generated and then does the following:

If there is exactly one bulletin generated at each time instance $0, \ldots, n-1$, then the receiver stacks the values of the information section of each bulletin into the vector:

$$\tilde{Y}_i^j[0] = \left[\tilde{y}_i^T[0]|\tilde{y}_i^T[1]|\ldots|\tilde{y}_i^T[n-1]\right]^T$$

where $i \in P$ is the first identifier in the routing section of class j. Otherwise, the receiver removes all bulletins in class j from its storage.

Step 3: The receiver picks a set $L \subseteq (P \cup V \cup E)$ of f_1 links and f_2 nodes[3] and removes from storage any bulletin in a class whose routing section contains at least 1 element in L.

Step 4: The receiver then checks whether the remaining bulletins are consistent (the consistency of bulletins will be defined afterwards). If the remaining bulletins are consistent, the state value that is consistent with all the remaining bulletins is the correct initial value of the state. Otherwise, it restores those removed bulletins in step (3), then goes to step (3) and picks another L.

[2]To do this the receiver should have a reasonable estimate of the size of the network.

[3]Note that the receiver does not need prior knowledge of the network since it can observe the identifiers of nodes and links, by analyzing the messages (including those attacked ones) it received.

Authorized licensed use limited to: UCLA Library. Downloaded on January 13,2022 at 02:57:08 UTC from IEEE Xplore. Restrictions apply.

The previous algorithm used the notion of consistency between bulletins. A set of bulletins generating $\tilde{Y}_i^j, i \in P$ in step (2), is said to be consistent if there exists a state $\tilde{x}[0]$ such that[4]:

$$\tilde{Y}_i^j[0] = \mathcal{O}_i\tilde{x}[0] \tag{4}$$

for each $\tilde{Y}_i^j[0]$ in step (2). In this case we also say the state $\tilde{x}[0]$ is consistent with these bulletins.[5]

Before proving correctness of the decoding algorithm we return to the example in Section V-A (see Fig. 2) to illustrate how the decoding algorithm works.

We consider a scalar system with trivial dynamics (i.e., $A = 1$ and $C_i = 1$) for all sensing nodes which yields $y_i = x$. We use the same network as in Fig. 2 in Section V-A. All messages are represented on the figure. Recall that a vertical line divides each bulletin into an information section and a routing section. In this example, we assume $x = 1$ hence $y_1 = y_2 = y_7 = 1$, and the attacked link changes the value of the information section from 1 to 2 whenever a bulletin passes through. Also, when node 8 generates a new bulletin, it places value 2 in the information section.

The receiver node 9 receives 9 bulletins in total, among which the first, fourth and eighth are attacked. Since the receiver knows that at most 1 link and 1 node are attacked, it selects a set composed of 1 link and 1 node and removes the bulletins containing them. For example, if the receiver picks link $3 \to 5$ and node 4, and then removes all bulletins whose routing section contains them, the only remaining bulletins are $2|8 \to 9$ and $1|7 \to 9$, which are inconsistent. Then by step 4 in the decoding algorithm, the receiver restores all removed messages and chooses another set. Since the combinations of 1 link and 1 node are finite, the receiver will eventually choose link $3 \to 4$ and node 8. Then, by removing all the bulletins whose routing section contains $3 \to 4$ and 8, the receiver concludes that the remaining bulletins are consistent and thus obtains the correct initial value $x[0]$ of the state.

C. Correctness of Decoding Algorithm

Lemma 3: If L, the set the receiver picks in step (3) of the decoding algorithm, contains all attacked links and attacked nodes, the remaining bulletins in step (3) of the algorithm are consistent and the initial value $x[0]$ of the state is the unique vector satisfying $\tilde{Y}_i^j[0] = \mathcal{O}_i x[0]$ with j ranging through the bulletins of the messages whose routing section contains no elements in L.

Proof: By Lemma (2), if all the attacked links and nodes are in L, all messages that might be incorrect are removed by the receiver. Hence by Lemma (1) the value of the information section $\tilde{y}_i$ of each bulletin whose routing section contains no elements of L satisfies $\tilde{y}_i[0] = C_i x[0]$, or equivalently, $\tilde{Y}_i^j[0] = \mathcal{O}_i x[0]$ since they are in the same class. In other words, the estimated $x[0]$ is consistent with all these bulletins.

To show that the solution $x[0]$ is unique, we assume, for the purpose of contradiction, the existence of a state $\tilde{x}[0]$, different from $x[0]$, and consistent with the bulletins whose routing section contains no elements in L, i.e., $\tilde{Y}_i^j[0] = \mathcal{O}_i\tilde{x}[0]$. We focus on the subset S of transmitting nodes defined by $S = \{w_i | \mathcal{O}_i(\tilde{x}[0] - x[0]) \neq 0\}$. By Assumption 1 in Section III, S is non-empty. Note that S is a critical set since the vector space spanned by $\tilde{x}[0] - x[0]$ is in the unobservable space of the set $P\backslash S$, i.e., for any sensing node $w_i \in P\backslash S$ we have $\mathcal{O}_i x[0] = 0$. But we also notice that, by assumption, for this critical set S, every mix cut H_S satisfies $L(H_S) > 2f_1$ or $N(H_S) > 2f_2$. This implies that by removing L from the network there should be at least 1 node $w_i \in S$ whose measurements $Y_i^j[0] = \mathcal{O}_i x[0]$ appears in the message sections of a certain bulletin during this time at the receiver without being altered. This establishes the contradiction since $Y_i^j[0] - \mathcal{O}_i\tilde{x}[0] = \mathcal{O}_i(x[0] - \tilde{x}[0]) \neq 0$. ■

Lemma 4: If there are bulletins with attacked measurements remaining in step 2 of the decoding algorithm, the remaining bulletins are inconsistent.

Proof: We follow an argument similar to the one used in the proof of Lemma (3). For the purpose of contradiction, we assume the remaining measurements are consistent, i.e., there exists a state $\tilde{x}[0]$ satisfying $Y_i^j[0] = \mathcal{O}_i\tilde{x}[0]$ for all remaining bulletins. Since there is at least 1 bulletin with altered measurements at the receiver, $\tilde{x}[0] \neq x[0]$. Still we focus on the non-empty set of sensing nodes $S = \{w_i | \mathcal{O}_i(\tilde{x}[0] - x[0]) \neq 0\}$, which is also a critical set as discussed above. By the assumption that for all remaining bulletins $Y_i^j[0] = \mathcal{O}_i\tilde{x}[0]$, no measurement from $w_i \in S$ arrives at the receiver without being altered. In other words, L, together with the actual set of attacked links and nodes M, forms a cut between S and the receiver. Since both L and M have at most f_1 links and f_2 nodes, we reach a contradiction with the fact that any cut has at least $2f_1$ links or $2f_2$ nodes. ■

The correctness of the algorithm is a natural consequence of Lemma (3) and Lemma (4). Moreover, since the proposed flooding protocol does not require relay nodes to transmit different messages to different out-neighbors, our algorithm applies both to wired and wireless networks.

VI. Necessity: Strategy of the Adversary

To show that the SSR problem is unsolvable if there exists a cut containing $2f_1$ links and $2f_2$ nodes or less, we prove that there always exists an attacking strategy for the adversary that prevents the adversary from correctly reconstructing the state.

The adversary's strategy is as follows:

Step 1: Find a critical set S and a mix cut H_S such that $L(H_S) \leq 2f_1$ and $N(H_S) \leq 2f_2$.

Step 2: Fabricate another state $\hat{x}[0]$ such that $x[0] - \hat{x}[0]$ is in the unobservable space of $P\backslash S$.

Step 3: Attack f_1 links and f_2 nodes in H_S. We denote by H_S^{ho} the set of non-attacked links and nodes, and H_S^{mal} the set of attacked links and nodes. For each link and node in H_S^{ho}, the attack assigns a link or a node in H_S^{mal} without repetition to simulate its behavior (which will be illustrated next).

[4]Again we note that k is the time index when the message is sent instead of received.

[5]The presence of bounded noise would not affect the results substantially since instead of checking equality (4) we would check instead $|\tilde{Y}_i^j[0] - \mathcal{O}_i\tilde{x}[0]| < \epsilon$ where ϵ is chosen based on the noise magnitude.

Authorized licensed use limited to: UCLA Library. Downloaded on January 13,2022 at 02:57:08 UTC from IEEE Xplore. Restrictions apply.

Therefore, each message passing through an attacked link (resp. node) is replaced with the message the honest link (resp. node) being simulated would have sent if the state was $\hat{x}$ instead of x.

To prove that such a strategy works for the adversary, we first notice that since S is a critical cut and the pair $(A, P\backslash S)$ is not observable, the state $\hat{x}$ described in step 2 is guaranteed to exist. Also the assignment in step 3 is feasible since there are more links and nodes in H_S^{mal} than in H_S^{ho}.

The receiver node is unable to distinguish the following 2 cases. **Case 1**: links and nodes in H_S^{mal} are attacked and the state is $x[0]$; and **Case 2**: links and nodes in H_S^{ho} are attacked and the state is $\hat{x}[0]$.

To prove this claim, we first notice that the measurements $Y_i^j[0]$ $(w_i \in P\backslash S)$ do not help distinguish between these two cases, since $\mathcal{O}_i(x[0] - \hat{x}[0]) = 0$ implies that $Y_i^j[0] = \mathcal{O}_i x[0] = \mathcal{O}_i \hat{x}[0] = \hat{Y}_i^j[0]$ for all $w_i \in P\backslash S$. Therefore, the removal of sensing nodes $P\backslash S$ and the corresponding links will not interfere with distinguishing these two cases.

Now we only focus on the cut H_S and the network from H_S to the receiver. We call each link and node in H_S a virtual sensing link and node respectively. The rest of the network is equivalent to the following: a set of virtual sensing nodes and links with at least half being attacked, and an error-free network connecting these virtual sources to the receiver. The correct information carried by virtual sources comes in pairs with an incorrect one (recall that the adversary assigns without repetition an attacked link and node to simulate the behavior of honest ones respectively), which is exactly the same except for the state x or $\hat{x}$ they encode.

In this case, if the receiver, somehow, decodes the correct message x, then in the scenario where everything is the same except that the adversary attacks H_S^{ho} using the same strategy, the receiver will conclude the state to be $\hat{x}[0]$. Therefore, the receiver cannot distinguish between these two cases.

Combining the results in Section V and VI we obtain Theorem 1.

VII. Conclusion

In this letter we investigated the SSR problem under an attack model that allows for attacks on sensors and also on communication. We provided necessary and sufficient conditions for the SSR problem to be solvable that generalize the existing results for the case of sensor only attacks and of communication only attacks on synchronous networks.

Acknowledgment

The authors would like to thank Dr. Gaurav Kumar Agarwal for helpful discussions.

References

[1] K. R. Davis *et al.*, "A cyber-physical modeling and assessment framework for power grid infrastructures," *IEEE Trans. Smart Grid*, vol. 6, no. 5, pp. 2464–2475, Sep. 2015.

[2] D. D'Auria and F. Persia, "A collaborative robotic cyber physical system for surgery applications," in *Proc. IEEE Int. Conf. Inf. Reuse Integr. (IRI)*, San Diego, CA, USA, Aug. 2017, pp. 79–83.

[3] A. A. Cárdenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems," in *Proc. 3rd Conf. Hot Topics Security (HOTSEC'08)*, 2008, pp. 1–6.

[4] S. Weerakkody, O. Ozel, Y. Mo, and B. Sinopoli, "Resilient control in cyber-physical systems: Countering uncertainty, constraints, and adversarial behavior," *Found. Trends Syst. Control*, vol. 7, nos. 1–2, pp. 1–252, 2019.

[5] P. Cheng, L. Shi, B. Sinopoli, "Guest editorial special issue on secure control of cyber-physical systems," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 1, pp. 1–3, Mar. 2017.

[6] J. Giraldo *et al.*, "A survey of physics-based attack detection in cyber-physical systems," *ACM Comput. Survey J.*, vol. 51, no. 4, pp. 1–36, 2018.

[7] Y. Mao, A. Mitra, S. Sundaram, and P. Tabuada, "When is the secure state-reconstruction problem hard?" in *Proc. 58st IEEE Conf. Decis. Control (CDC)*, Nice, France, 2019, pp. 5368–5373.

[8] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Trans. Autom. Control*, vol. 58, no. 11, pp. 2715–2729, Nov. 2013.

[9] C.-Z. Bai, F. Pasqualetti, and V. Gupta, "Data-injection attacks in stochastic control systems: Detectability and performance tradeoffs," *Automatica*, vol. 82, pp. 251–260, Aug. 2017.

[10] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Trans. Autom. Control*, vol. 59, no. 6, pp. 1454–1467, Jun. 2014.

[11] S. Z. Yong, M. Q. Foo, and E. Frazzoli, "Robust and resilient estimation for cyber-physical systems under adversarial attacks," in *Proc. Amer. Control Conf. (ACC)*, Boston, MA, USA, Jul. 2016, pp. 308–315.

[12] Y. Shoukry and P. Tabuada, "Event-triggered state observers for sparse sensor noise/attacks," *IEEE Trans. Autom. Control*, vol. 61, no. 8, pp. 2079–2091, Aug. 2016.

[13] Y. Shoukry, P. Nuzzo, A. L. Sangiovanni-Vincentelli, S. A. Seshia, G. J. Pappas, and P. Tabuada, "SMC: Satisfiability modulo convex programming," *Proc. IEEE*, vol. 106, no. 9, pp. 1655–1679, Sep. 2018.

[14] A. Tiwari *et al.*, "Safety envelope for security," in *Proc. 3rd Int. Conf. High Confidence Netw. Syst.*, 2014, pp. 85–94.

[15] L. An and G. Yang, "State estimation under sparse sensor attacks: A constrained set partitioning approach," *IEEE Trans. Autom. Control*, vol. 64, no. 9, pp. 3861–3868, Sep. 2019.

[16] Y. Mo and E. Garone, "Secure dynamic state estimation via local estimators," in *Proc. 55st IEEE Conf. Decis. Control (CDC)*, Las Vegas, NV, USA, Dec. 2016, pp. 5073–5078.

[17] L. An and G. Yang, "Secure state estimation against sparse sensor attacks with adaptive switching mechanism," *IEEE Trans. Autom. Control*, vol. 63, no. 8, pp. 2596–2603, Aug. 2018.

[18] A. Mitra and S. Sundaram, "Byzantine-resilient distributed observers for LTI systems," 2018. [Online]. Available: arXiv:1802.09651.

[19] A. Mitra, J. A. Richards, S. Bagchi, and S. Sundaram, "Resilient distributed state estimation with mobile agents: Overcoming byzantine adversaries, communication losses, and intermittent measurements," *Auton. Robots*, vol. 43, no. 3, pp. 743–768, Mar. 2019.

[20] R. Yeung and N. Cai, "Network error correction, I: Basic concepts and upper bounds," *Commun. Inf. Syst.*, vol. 6, pp. 19–35, Jan. 2006.

[21] L. Czap, C. Fragouli, V. M. Prabhakaran, and S. Diggavi, "Secure network coding with erasures and feedback," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1667–1686, Apr. 2015.

[22] T. Li, B. K. Dey, S. Jaggi, M. Langberg, and A. D. Sarwate, "Quadratically constrained channels with causal adversaries," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Vail, CO, USA Jun. 2018, pp. 621–625.

[23] R. Koetter and F. R. Kschischang, "Coding for errors and erasures in random network coding," *IEEE Trans. Inf. Theory*, vol. 54, no. 8, pp. 3579–3591, Aug. 2008.

[24] T. Etzion and A. Vardy, "Error-correcting codes in projective space," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 1165–1173, Feb. 2011.

[25] M. S. Chong, M. Wakaiki, and J. P. Hespanha, "Observability of linear systems under adversarial attacks," in *Proc. Amer. Control Conf. (ACC)*, Chicago, IL, USA, Jul. 2015, pp. 2439–2444.

[26] L. Su and N. H. Vaidya, "Fault-tolerant multi-agent optimization: Optimal iterative distributed algorithms," in *Proc. ACM Symp. Principles Distrib. Comput.*, 2016, pp. 425–434.

[27] C. Fragouli and E. Soljanin, "Network coding fundamentals," *Found. Trends Netw.*, vol. 2, no. 1, pp. 1–133, 2007.

Authorized licensed use limited to: UCLA Library. Downloaded on January 13,2022 at 02:57:08 UTC from IEEE Xplore. Restrictions apply.