Automatica 116 (2020) 108920

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Securing state reconstruction under sensor and actuator attacks: Theory and design[☆]

Mehrdad Showkatbakhsh[a], Yasser Shoukry[b], Suhas N. Diggavi[a], Paulo Tabuada[a,*]

[a] *Electrical & Computer Engineering Department, UCLA, Los Angeles, CA, United States of America*
[b] *Department of Electrical Engineering and Computer Science, University of California, Irvine, CA, United States of America*

ARTICLE INFO

Article history:
Received 21 December 2018
Received in revised form 8 September 2019
Accepted 14 February 2020
Available online xxxx

Keywords:
Cyber–physical security
State reconstruction
Security monitoring

ABSTRACT

This paper discusses the problem of reconstructing the state of a linear time invariant system when some of its actuators and sensors are compromised by an adversarial agent. In the model considered in this paper, the adversarial agent attacks an input (output) by manipulating its value arbitrarily, i.e., we impose no constraints (statistical or otherwise) on how control commands (sensor measurements) are changed by the adversary other than a bound on the number of attacked actuators and sensors In the first part of this paper, we introduce the notion of sparse strong observability and we show that is a necessary and sufficient condition for correctly reconstructing the state despite the considered attacks. In the second half of this work, we propose an observer to harness the complexity of this intrinsically combinatorial problem, by leveraging satisfiability modulo theory solving. Numerical simulations illustrate the effectiveness and scalability of our observer.

1. Introduction

Cyber–Physical Systems (CPS) are characterized by the tight interconnection of cyber and physical components. CPS are not only prone to actuator and sensor failures but also to adversarial attacks on the control and sensing modules. Security of CPS is no longer restricted to the cyber domain, and recent incidents such as the StuxNet malware (Langner, 2011) and the security flaws reported on modern cars (Greenberg, 2015; Kelion, 2016) motivated the recent interest in security of CPS, (see for example, Amin, Schwartz, & Hussain, 2013; Cárdenas, Amin, & Sastry, 2008; Mo et al., 2012; Sundaram, Pajic, Hadjicostis, Mangharam, & Pappas, 2010 and references therein). During the last decade, a number of security problems have been tackled by the control community, *e.g.*, denial-of-service (De Persis & Tesi, 2015; Gupta, Langbort, & Basar, 2010; Senejohnny, Tesi, & De Persis, 2016; Zhu & Martinez, 2014), replay attacks (Mo & Sinopoli, 2009), man-in-the-middle attacks (Smith, 2015), false data injection (Mo, Garone, Casavola, & Sinopoli, 2010), etc.

This paper addresses the problem of state reconstruction when several sensors *and* actuators are under attack. We broadly refer to state reconstruction in the adversarial environment as secure state reconstruction. Our attack model is quite general and we impose no constraints on the magnitude, statistical properties, or temporal characteristics of the signals manipulated by the adversary other than a bound on the number of attacked actuators and sensors.

The problem of secure state reconstruction has been investigated by the control community over the past decade (Giraldo et al., 2018). In one line of work, the problem of state reconstruction and control under sensor attacks is investigated and the authors derived necessary and sufficient conditions under which reconstruction and stabilization are possible (Fawzi, Tabuada, & Diggavi, 2014). Shoukry and Tabuada (2016) further refined this condition and called it sparse observability. Chong, Wakaiki, and Hespanha (2015) found an equivalent condition for continuous-time systems and called it observability under attack. Nakahira and Mo (2015) investigated a similar problem while allowing for asymptotic reconstruction of the state rather than reconstruction in finite time. The authors relaxed the sparse observability condition to sparse detectability and showed it is a necessary and sufficient condition for asymptotic reconstruction. Noisy versions of this problem have also been investigated (Bai, Gupta and Pasqualetti, 2017; Bai, Pasqualetti and Gupta, 2017; Mishra,

[☆] The work of the first, third, and fourth authors was partially supported by the National Science Foundation grants 1705135 and 1740047, by the UC-NL grant LFR-18-548554, and by the Army Research Laboratory under Cooperative Agreement W911NF-17-2-0196. The work of the second author was supported by the NSF grants 2002405 and 2013824. The material in this paper was partially presented at the 56th IEEE Conference on Decision and Control, December 12–15, 2017, Melbourne, Australia. This paper was recommended for publication in revised form by Associate Editor Vijay Gupta under the direction of Editor Christos G. Cassandras.

[*] Corresponding author.
E-mail addresses: mehrdadsh@ucla.edu (M. Showkatbakhsh), yshoukry@uci.edu (Y. Shoukry), suhas@ee.ucla.edu (S.N. Diggavi), tabuada@ee.ucla.edu (P. Tabuada).

https://doi.org/10.1016/j.automatica.2020.108920
0005-1098/© 2020 Elsevier Ltd. All rights reserved.

Shoukry, Karamchandani, Diggavi, & Tabuada, 2017; Mo, Chabukswar, & Sinopoli, 2014; Mo & Sinopoli, 2016). In particular, Mishra et al. (2017) derived the optimal solution for Gaussian noise.

In this paper, we solve the more general problem of *actuator and sensor* attacks that includes, as a special case, sensor attacks. Under the attack model in which an adversary can only target a bounded number of actuators and sensors, state reconstruction is intrinsically a combinatorial problem. In the case where only sensors are attacked, Shoukry et al. (2017) proposed a novel secure state observer using the Satisfiability Modulo Theory (SMT) paradigm, called IMHOTEP-SMT, that offers a computationally efficient solution to this problem. In this paper, we generalize the SMT approach to handle sensor *and* actuator attacks.

In another line of work, the problem of secure state reconstruction has been studied when the exact model of the system is not available (Pajic et al., 2014; Yong, Foo, & Frazzoli, 2016). Tiwari et al. (2014) proposed an online learning method by building so-called safety envelopes as it receives attack-free data to detect abnormality in the data when the system is prone to attacks. In Showkatbakhsh, Tabuada, and Diggavi (2016a, 2016b) the authors considered system identification under sensors attacks. In all of these works, the adversarial agent is restricted to only attacking sensors.

Pasqualetti, Dorfler, and Bullo (2013) is one of the few references considering attacks to both sensor and actuators. They studied the problem of attack detection and identification and related undetectable and unidentifiable attacks to the zero-dynamics of the underlying system. They also proposed an attack identification mechanism consisting of a number of fault-monitor filters that provide formal guarantees for the existence of an attack. The number of filters, however, grows exponentially with the number of attacked sensors/actuators, and therefore hinders scalability. In another work (Sandberg & Teixeira, 2016), the authors investigated detectability and identifiability of attacks in the presence of disturbances and generalized the concept of security index to dynamical systems. The proposed algorithms are inherently combinatorial and do not scale well with the number of attacked sensors and actuators. In this paper, by leveraging the SMT paradigm, we design a state observer that scales well with the number of sensors and actuators.

In a recent work (Harirchi & Ozay, 2016), Harirchi et al. proposed a novel fault detection approach using techniques from model invalidation. The authors pursued a worst-case scenario approach and therefore their framework is suitable for security. However, necessary and sufficient conditions for state reconstruction in a general adversarial setting were not investigated in Harirchi and Ozay (2016). In this paper, we precisely characterize the class of systems, by providing necessary and sufficient conditions, for which state reconstruction is possible despite sensor and/or actuator attacks.

The contributions of this paper can be summarized as follows:

- We introduce the notion of sparse strong observability by drawing inspiration from sparse observability (Fawzi et al., 2014; Shoukry & Tabuada, 2016) and the classical notion of strong observability (Hautus, 1983). We show that sparse strong observability is necessary and sufficient to correctly reconstruct the state in the presence of actuator and sensor attacks.
- We propose an observer by leveraging the SMT approach to harness the exponential complexity of the secure state reconstruction problem. Our observer consists of two blocks interacting iteratively until the true state is found (see Section 4 for a detailed explanation of the observer's architecture).
- We propose two methods to further decrease the running time of the aforementioned algorithm by reducing the number of iterations of the observer. The first method exploits heuristics that can be efficiently computed at each iteration. The second method is inspired by the QUICKXPLAIN algorithm (Junker, 2001) that efficiently finds an irreducibly inconsistent set (see Section 4 for a detailed discussion on the aforementioned methods). We illustrate the scalability of our proposed observer by several numerical simulations.

A preliminary version of some of the results in this paper was presented in Showkatbakhsh, Shoukry, Chen, Diggavi, and Tabuada (2017) where we introduced the notion of sparse strong observability and drew the connection to secure state reconstruction. However, the formal proofs were not provided due to space limitations. Furthermore, we propose in this paper a new observer that outperforms the observer introduced in Showkatbakhsh et al. (2017).

This paper is organized as follows. Section 2 introduces notation followed by the attack model and the precise problem formulation. In Section 3, we introduce the notion of sparse strong observability and relate this notion to the problem of state reconstruction when some of the inputs and outputs are under adversarial attacks. This section concludes with the main theoretical contribution of this paper that is Theorem 8. Section 4 is devoted to designing an observer by exploiting the SMT paradigm. Section 5 provides the simulation results followed by Section 6 that concludes the paper.

2. Problem definition

2.1. Notation

We denote the sets of real, natural and binary numbers by $\mathbb{R}$, $\mathbb{N}$ and $\mathbb{B}$. We represent vectors and real numbers by lowercase letters, such as u, x, y, and matrices with capital letters, such as A. Given a vector $x \in \mathbb{R}^n$ and a set $O \subseteq \{1, \dots, n\}$, we use $x|_O$ to denote the vector obtained from x by removing all elements except those indexed by the set O. Similarly, for a matrix $C \in \mathbb{R}^{n_1 \times n_2}$ we use $C|_{(O_1,O_2)}$ to denote the matrix obtained from C by eliminating all rows and columns except the ones indexed by O_1 and O_2, respectively, where $O_i \subseteq \{1, \dots, n_i\}$ with $n_i \in \mathbb{N}$ for $i \in \{1, 2\}$. In order to simplify the notation, we use $C|_{(.,O_2)} := C|_{(\{1,\dots,n_1\},O_2)}$ and $C|_{(O_1,.)} := C|_{(O_1,\{1,\dots,n_2\})}$. We denote the complement of O by $\overline{O} := \{1, \dots, n\} \setminus O$. We use the notation $\{x(t)\}_{t=0}^{T-1}$ to denote the sequence $x(0), \dots, x(T-1)$, and we drop the sub(super)scripts whenever it is clear from the context.

A Linear Time Invariant (LTI) system is described by the following equations:

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

where $u(t) \in \mathbb{R}^m$, $x(t) \in \mathbb{R}^n$ and $y(t) \in \mathbb{R}^p$ are the input, state and output variables, respectively, $t \in \mathbb{N} \cup \{0\}$ denotes time, and A, B, C and D are system matrices with appropriate dimensions. We use (A, B, C, D) to denote the system described by (1). The order of an LTI system is defined as the dimension of its state space. A trajectory of the system consists of an input sequence with its corresponding output sequence. For an LTI system:

$$\mathcal{O}_{(A,C)} := \begin{bmatrix} C^T & A^T C^T & \dots & (A^T)^{n-1} C^T \end{bmatrix}^T, \tag{2}$$

$$\mathcal{N}_{(A,B,C,D)} := \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \vdots & & \ddots & \\ CA^{n-2}B & CA^{n-3}B & \dots & D \end{bmatrix}, \tag{3}$$

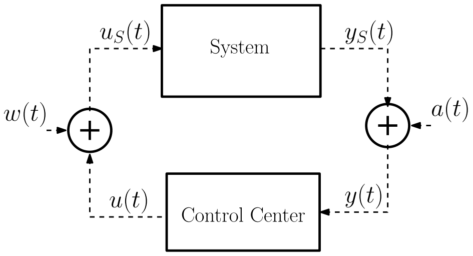

Fig. 1. The generic attack model considered in this paper.

are the *observability* and *invertibility* matrices, respectively, where n is the order of the underlying system. In this paper, we often work with subsets of inputs and outputs. For a subset of outputs $\Gamma_y \subseteq \{1,\ldots,p\}$, we use the notation $\mathscr{O}_{\Gamma_y} := \mathscr{O}_{(A,C|_{(\Gamma_y,.)})}$ to denote the observability matrix of outputs in the set Γ_y. For a set of inputs $\Gamma_u \subseteq \{1,\ldots,m\}$, we use the notation $\mathscr{N}_{\Gamma_u \to \Gamma_y}$ to denote $\mathscr{N}_{(A,B_{(.,\Gamma_u)},C_{(\Gamma_y,.)},D_{(\Gamma_y,\Gamma_u)})}$. For $x \in \mathbb{R}^n$, we define its support set as the set of indices of its non-zero components, denoted by $\text{supp}(x)$. Similarly, we define the support of the sequence $\{x(t)\}$ as $\text{supp}(\{x(t)\}) := \cap_t \text{supp}(x(t))$. The observer proposed in this paper uses batches of inputs and outputs in order to reconstruct the state. We reserve capital bold letters to denote these batches:

$$\mathbf{Y}^\tau(t) := \begin{bmatrix} y(t-\tau+1)^T & \ldots & y(t)^T \end{bmatrix}^T, \tag{4}$$

$$\mathbf{U}^\tau(t) := \begin{bmatrix} u(t-\tau+1)^T & \ldots & u(t)^T \end{bmatrix}^T, \tag{5}$$

where $\tau \leq n$. Whenever τ is the order of the underlying system, we may drop the superscript for ease of notation. For a subset of outputs (inputs), denoted by $\Gamma_y \subseteq \{1,\ldots,p\}$ ($\Gamma_u \subseteq \{1,\ldots,m\}$), we use the notation $\mathbf{Y}^\tau|_{\Gamma_y}(t)$ ($\mathbf{U}^\tau|_{\Gamma_u}(t)$) for the batches of length τ that only consists of outputs (inputs) in the set Γ_y (Γ_u). For a vector $x \in \mathbb{R}^n$, we denote an arbitrary norm, l_2-norm and l_1-norm of x by $\|x\|$, $\|x\|_2$ and $\|x\|_1$.

2.2. System and attack model

This work is concerned with the problem of state reconstruction of LTI systems. We consider the scenario in which sensors and actuators are both prone to adversarial attacks. The ultimate goal is to reconstruct the state despite these attacks. In this part, we define the attack model and conclude this section with the precise problem statement. The system S, is described by the following equations:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu_S(t), \\ y_S(t) &= Cx(t) + Du_S(t). \end{aligned} \tag{6}$$

Without loss of generality we assume $\begin{bmatrix} B^T & D^T \end{bmatrix}^T$ to be of full column rank.

Each actuator (sensor) corresponds to one input (output) and we use input (output) instead of actuator (sensor) in the rest of this paper. In this set up the adversary can attack both inputs and outputs. We model these attacks by the additive terms:

$$\begin{cases} u_S(t) &= u(t) + w(t) \\ y(t) &= y_S(t) + a(t), \end{cases} \tag{7}$$

where $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$ are the controller-designed input and the observed output, respectively, and $w(t) \in \mathbb{R}^m$ and $a(t) \in \mathbb{R}^p$ are signals injected by the malicious agent. We note that an actuator attack changes the plant model since u_S in (6) is given by (7) and may be different from u. However, a sensor attack does not. Instead, a sensor attack changes the input y to the controller that may become different from y_S according to (7). In the rest of this paper, we refer to the signals $(w(t), a(t))$ as the attack of the adversarial agent. We use the subscript S for signals that directly come from/to the system. The controller can only observe $y(t)$ and compute the input $u(t)$. This generic attack model is depicted in Fig. 1.

When the adversary attacks an input (output) it can change its value to any arbitrary number without explicitly revealing its presence. The only limitation that we impose on the power of the malicious agent is the maximal number of inputs and outputs that can be attacked.

Assumption 1 (*Bound on the Number of Attacks*). The number of inputs and outputs under attack is bounded by r and s, respectively.

Therefore, the malicious agent can attack a subset of inputs and outputs denoted by $\Gamma_u \subseteq \{1,\ldots,m\}$ and $\overline{\Gamma}_y \subseteq \{1,\ldots,p\}$,[1] respectively, with $|\Gamma_u| \leq r$ and $|\overline{\Gamma}_y| \leq s$, such that $\text{supp}(\{w(t)\}) \subseteq \Gamma_u$ and $\text{supp}(\{a(t)\}) \subseteq \overline{\Gamma}_y$. Note that these sets are not known to the controller and only upper bounds on their cardinality are given. Once the adversary chooses these sets, inputs and outputs outside these sets remain attack-free. This assumption is realistic when the time it takes for the adversarial agent to attack new inputs and outputs is large compared to the time scale of the system.

We now precisely define the main problem we tackle in this paper.

Problem 2 (*Secure State Reconstruction*). For the linear system defined by (6) under the attack model defined by (7), what are necessary and sufficient conditions under which the state of the compromised system (6) can be reconstructed with bounded delay?

It is well-known that the secure state reconstruction problem, when only outputs are under adversarial attacks, is combinatorial and belongs to the class of *NP-hard* problems (Pasqualetti et al., 2013; Shoukry et al., 2017). Therefore we are motivated to design an observer that harness the complexity of this problem.

Problem 3 (*Secure Observer Design*). Assuming conditions in Problem 2 are satisfied, how can we design an observer that reconstructs the state of the compromised system?

3. Conditions for secure state reconstruction

In this section, we solve Problem 2, i.e., we provide conditions on the system described by (6) under which state reconstruction (with bounded delay) is possible. We first develop the notion of sparse strong observability. This section concludes with Theorem 8 that relates this notion to the solution of Problem 2.

In the absence of attacks, the problem of reconstructing the state of a system without the knowledge of some of its inputs has been investigated and its solution characterized by the notion of strong observability (Hautus, 1983). In the case of an attack, we can think of unknown inputs as the signals injected by an adversary in a sensor attack. We now recall the notion of strong observability.

Definition 4 (*Strong Observability*). An LTI system, given by (1), is called strongly observable if for any initial state $x(0) \in \mathbb{R}^n$ and any input sequence $\{u(t) \in \mathbb{R}^m\}_{t=0}^{\infty}$ there exists an integer $\tau \in \mathbb{N} \cup \{0\}$ such that $x(0)$ can be uniquely recovered from $\{y(t)\}_{t=0}^{\tau}$.

[1] For ease of exposition, we use Γ_u to denote under-attack inputs while using Γ_y for the set of attack-free outputs, i.e., the set of under-attack outputs is represented by $\overline{\Gamma}_y := \{1,\ldots,p\} \setminus \Gamma_y$ in this paper.

Note that τ is always upper-bounded by the order of the system. Linearity of the system implies the following lemma.

Lemma 5. *An LTI system, given by (1), is strongly observable if and only if $y(t) = 0\ \forall t \in \mathbb{N} \cup \{0\}$ implies that $x(0) = 0$.*

Proof. Please refer to Appendix.

It is straightforward to conclude the following corollary.

Corollary 6. *An LTI system, given by (1), is not strongly observable if and only if there exist a non-zero initial state and an input sequence such that $y(t) = 0$ for $t \in \mathbb{N} \cup \{0\}$.*

Proof. Follows directly from Lemma 5.

It is well-understood that when the adversary is restricted to attacking outputs, state reconstruction is possible only if there is enough redundancy in the outputs of the system. This redundancy can be stated in terms of observability of the system while removing a number of outputs. This property has been formalized in Fawzi et al. (2014) and is called sparse observability (Shoukry & Tabuada, 2016). By analogy with sparse observability, we define the notion of (r, s)-sparse strong observability as follows:

Definition 7 (*(r, s)-sparse Strong Observability*)**.** An LTI system (A, B, C, D) with m inputs and p outputs, given by (1), is (r, s)-sparse strongly observable if for any $\Gamma_u \subseteq \{1, \ldots, m\}$ and $\Gamma_y \subseteq \{1, \ldots, p\}$ with $|\Gamma_u| \leq r$ and $|\Gamma_y| \geq p - s$, the system $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ is strongly observable.

Note that in Definition 7, the value of r and s are upper bounded by the number of inputs and outputs, respectively. This modified notion of strong observability is the key for formalizing redundancy across inputs and outputs. We show that a necessary and sufficient condition for secure state reconstruction can be stated using this property. Note that $(0, s)$-sparse strong observability is equivalent to the notion of s-sparse observability that was introduced before in the literature (Fawzi et al., 2014; Mishra et al., 2017; Shoukry & Tabuada, 2016). The following theorem is the main theoretical result in this paper.

Theorem 8. *Consider an LTI system, given by (1), subject to sensor and actuator attacks according to the attack model (7) and let the number of attacked inputs and outputs be bounded by r and s, respectively. The state of the LTI system can be reconstructed (possibly with delay) if and only if the LTI system is $(2r, 2s)$-sparse strongly observable.*

Remark 9. It is worth mentioning that the maximum number of attacked outputs, s, cannot be greater than $\lfloor \frac{p}{2} \rfloor$ and it is an inherent limitation of LTI systems with p outputs (Fawzi et al., 2014). However the maximum number of attacked inputs is not inherently restricted by $\lfloor \frac{m}{2} \rfloor$ and can take values up to m, depending on the specific system under the consideration.

Remark 10. Pasqualetti et al. (2013) addressed the problem of attack detection and identification in the presence of adversarial inputs and outputs for continuous-time LTI systems. They showed that attack identification is possible if and if for any $\Gamma_u \subseteq \{1, \ldots, m\}$ and $\Gamma_y \subseteq \{1, \ldots, p\}$ with $|\Gamma_u| \leq 2r$ and $|\Gamma_y| \geq p - 2s$, the system $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ does not have any invariant zeros.

If one is able to correctly reconstruct the state despite an attack, then using the dynamics we can also reconstruct the signals injected by the adversary. Hence, solvability of the secure state reconstruction problem implies attack detectability and identifiability. Conversely, if one is able to reconstruct the signals injected by the adversary, then it is possible to reconstruct the state. Hence, the problem of reconstructing the attacks and the problem of reconstructing the state are equivalent. Technically, the characterization of attack identifiability in Pasqualetti et al. (2013) is based on the absence of invariant zeros and it is known (see, for example Theorem 1.8 in Hautus, 1983) that strongly observable LTI systems do not have invariant zeros. Therefore, an alternative proof of Theorem 8 could be given in terms of attack identifiability. Here, however, we give a direct proof without resorting to attack identifiability.

Proof. First we show that $(2r, 2s)$-sparse strong observability is a sufficient condition for correctly reconstructing the state. For the sake of the contradiction, assume that the state cannot be reconstructed, i.e., there exist two different (initial) states, denoted by $x^{(1)}$ and $x^{(2)}$, that cannot be distinguished under this attack model. More precisely, there exist two attack strategies that will lead to the same exact (observed) trajectories. We reserve superscripts $.^{(1)}$ and $.^{(2)}$ for variables across those scenarios. Let us denote the adversarial additive terms by $\{w^{(1)}(t)\}$, $\{a^{(1)}(t)\}$ and $\{w^{(2)}(t)\}$, $\{a^{(2)}(t)\}$. We represent the corresponding inputs and outputs of the system by $\{u_S^{(1)}(t)\}$, $\{y_S^{(1)}(t)\}$ and $\{u_S^{(2)}(t)\}$, $\{y_S^{(2)}(t)\}$, and the common (corrupted) measured output and the controller input sequences are denoted by $\{y(t)\}$ and $\{u(t)\}$, respectively.

By the assumption of the attack model (7), there exist $\Gamma_u^{(i)}$, $\overline{\Gamma}_y^{(i)}$ for $i \in \{1, 2\}$ with bounded cardinality such that:

$$\mathrm{supp}(\{w^{(i)}(t)\}) \subseteq \Gamma_u^{(i)},\ \mathrm{supp}(\{a^{(i)}(t)\}) \subseteq \overline{\Gamma}_y^{(i)}, \tag{8}$$

for $i \in \{1, 2\}$. Note that:

$$\begin{cases} u_S^{(1)}(t) = u(t) + w^{(1)}(t) \\ u_S^{(2)}(t) = u(t) + w^{(2)}(t), \end{cases} \tag{9}$$

where $u(t)$ is the controller designed input. Therefore:

$$\begin{aligned} \mathrm{supp}(\{u_S^{(1)}(t) - u_S^{(2)}(t)\}) &= \mathrm{supp}(\{w^{(1)}(t) - w^{(2)}(t)\}) \\ &\subseteq \Gamma_u^{(1)} \cup \Gamma_u^{(2)}. \end{aligned} \tag{10}$$

Similarly, it is straightforward to conclude the inclusion $\mathrm{supp}(\{y_S^{(1)}(t) - y_S^{(2)}(t)\}) \subseteq \overline{\Gamma}_y^{(1)} \cup \overline{\Gamma}_y^{(2)}$. We are ready to reach the contradiction. The underlying system is LTI, thus the input sequence $\{u_S^{(1)}(t) - u_S^{(2)}(t)\}$ with the initial state $x^{(1)} - x^{(2)}$ generates the output sequence $\{y_S^{(1)}(t) - y_S^{(2)}(t)\}$. The underlying system is $(2r, 2s)$-sparse strongly observable so the sub-system $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ is strongly observable for any $|\Gamma_u| = 2r$ and $|\Gamma_y| = p - 2s$. Let us choose Γ_u and Γ_y as any set of $2r$ inputs and $p - 2s$ outputs such that:

$$\Gamma_u^{(1)} \cup \Gamma_u^{(2)} \subseteq \Gamma_u, \quad \Gamma_y \subseteq \Gamma_y^{(1)} \cap \Gamma_y^{(2)}. \tag{11}$$

Note that $\{y_S^{(1)}(t)|_{\Gamma_y} - y_S^{(2)}(t)|_{\Gamma_y}\}$ is a zero sequence, hence by Lemma 5 we conclude that the corresponding initial state $(x^{(1)} - x^{(2)})$ is zero, which contradicts the assumption of $x^{(1)} \neq x^{(2)}$. Now we prove that $(2r, 2s)$-sparse strong observability is a necessary condition. For the sake of contradiction, suppose that the system described by (6) is not $(2r, 2s)$-sparse strongly observable, however, reconstructing the state (possibly with delays) is still possible. We construct two system trajectories with different (initial) states that have exactly the same input and output sequences under suitable attack strategies (additive terms). This implies that reconstructing the correct state is indeed impossible thereby establishing the desired contradiction.

By the assumption of the contradiction, the underlying system is not $(2r, 2s)$-sparse strongly observable, so there exist subsets of inputs and outputs denoted by Γ_u with $|\Gamma_u| = 2r$ and Γ_y with

$|\Gamma_y| = p - 2s$, respectively, such that $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ is not strongly observable. Corollary 6 implies that there exist an initial condition Δx and an input sequence $\{\Delta u(t)\}$ (with its support lying inside Γ_u) that generates an output sequence $\{\Delta y(t)\}$ with $\text{supp}(\{\Delta y(t)\}) \subseteq \overline{\Gamma}_y$. One can rewrite $\Delta u(t)$ and $\Delta y(t)$ as sum of two sparse signals, more precisely:

$$\Delta u(t) = \Delta u^{(1)}(t) + \Delta u^{(2)}(t), \tag{12}$$

$$\Delta y(t) = \Delta y^{(1)}(t) + \Delta y^{(2)}(t), \tag{13}$$

where cardinality of $\text{supp}(\{\Delta u^{(i)}(t)\})$ and $\text{supp}(\{\Delta y^{(i)}(t)\})$ are upper-bounded by r and s for $i \in \{1, 2\}$, respectively. For example, we can rewrite $\overline{\Gamma}_y = \overline{\Gamma}_y^{(1)} \cup \overline{\Gamma}_y^{(2)}$ where $|\overline{\Gamma}_y^{(i)}| \leq s$ for $i \in \{1, 2\}$. Then we define:

$$\begin{cases} \Delta y^{(i)}(t)|_{\overline{\Gamma}_y^{(i)}} & := \Delta y(t)|_{\overline{\Gamma}^{(i)}} \\ \Delta y^{(i)}(t)|_{\Gamma_y^{(i)}} & := 0, \end{cases} \quad \text{for} \quad i \in \{1, 2\}.$$

Now consider the following two different trajectories of the system:

$$\begin{cases} u_S^{(1)}(t) & = \Delta u(t) \\ y_S^{(1)}(t) & = \Delta y(t), \end{cases} \quad \begin{cases} u_S^{(2)}(t) & = 0 \\ y_S^{(2)}(t) & = 0, \end{cases} \tag{14}$$

with their initial states:

$$\begin{cases} x^{(1)}(0) & = \Delta x \\ x^{(2)}(0) & = 0, \end{cases} \tag{15}$$

and their corresponding attack strategies:

$$\begin{cases} w^{(1)}(t) & = \Delta u^{(1)}(t) \\ a^{(1)}(t) & = -\Delta y^{(1)}(t), \end{cases} \quad \begin{cases} w^{(2)}(t) & = -\Delta u^{(2)}(t) \\ a^{(2)}(t) & = \Delta y^{(2)}(t). \end{cases} \tag{16}$$

It is straightforward to verify that $\{y^{(1)}(t)\} = \{y^{(2)}(t)\}$ and $\{u^{(1)}(t)\} = \{u^{(2)}(t)\}$, i.e., under the attack model (7) the controlled inputs and the observed outputs are exactly the same for both trajectories while having different states, therefore the proof is complete.

4. Secure observer design

In this section, we seek solutions to Problem 3. In the first part, we explain the intuition behind the proposed algorithm that reconstructs the state despite attacks on inputs and outputs. We give formal guarantees that the algorithm reconstructs the state correctly. In the second part, we introduce the observer by leveraging the SMT paradigm followed by two methods that enhance the run time of state reconstruction.

Based on the attack model (7), the input to the system is decomposed into two additive terms, the controller-designed input $u(t)$ and the adversarial input $w(t)$. The underlying system (6) is linear and therefore we can easily exclude the effect of the controller-designed input from the output by subtracting its effect. Hence, without loss of generality we assume that the true $u(t)$ is zero.

The proposed algorithm is based on the following proposition.

Proposition 11. *Consider an LTI system, given by (1), assume it is $(2r, 2s)$-sparse strongly observable, and that the number of attacked inputs and outputs is bounded by r and s, respectively. Given any subset of inputs and outputs denoted by Γ_u and Γ_y with $|\Gamma_u| \leq r$ and $|\Gamma_y| \geq p - s$, the first statement below implies the second:*

(1) There exist $\hat{\mathbf{U}} \in \mathbb{R}^{n|T|}$ and $\hat{x} \in \mathbb{R}^n$ such that

$$\mathbf{Y}|_{\Gamma_y}(t) = \mathscr{O}_{\Gamma_y}\hat{x} + \mathscr{N}_{\Gamma_u \to \Gamma_y}\hat{\mathbf{U}}. \tag{17}$$

(2) The reconstructed state $\hat{x}$, is equal to the actual state of the LTI system at time $t-n+1$, $x(t-n+1)$, where n is the order of the LTI system.

Remark 12. The LTI system is $(2r, 2s)$-sparse strongly observable therefore $(A, B_{(.,\Gamma_u)}, C_{(\Gamma_y,.)}, D_{(\Gamma_y,\Gamma_u)})$ is strongly observable. If (17) has a solution, then $\hat{x}$ would be the unique solution for x (see section III-B of Yoshikawa & Bhattacharyya, 1975).

Proof. Let us denote the set of attack-free outputs and under-attack inputs by Γ_y^* and Γ_u^*. At most s outputs are under attack, therefore $|\Gamma_y \cap \Gamma_y^*| \geq p - 2s$. Note that $\mathbf{Y}|_{\Gamma_y \cap \Gamma_y^*}$ can be written as follows:

$$\begin{aligned} \mathbf{Y}|_{\Gamma_y \cap \Gamma_y^*} =& O_{\Gamma_y \cap \Gamma_y^*} x(t-n+1) \\ & + \mathscr{N}_{\Gamma_u \to \Gamma_y \cap \Gamma_y^*}\mathbf{W}|_{\Gamma_u} + \mathscr{N}_{\Gamma_u^* \setminus \Gamma_u \to \Gamma_y \cap \Gamma_y^*}\mathbf{W}|_{\Gamma_u^* \setminus \Gamma_u}, \end{aligned} \tag{18}$$

where $\mathbf{W}$ is the sequence of inputs used to generate $\mathbf{Y}$. On the other hand, we can rewrite (17) by taking only outputs in $\Gamma_y \cap \Gamma_y^*$:

$$\mathbf{Y}|_{\Gamma_y \cap \Gamma_y^*} = O_{\Gamma_y \cap \Gamma_y^*}\hat{x} + \mathscr{N}_{\Gamma_u \to \Gamma_y \cap \Gamma_y^*}\hat{\mathbf{U}} + \mathscr{N}_{\Gamma_u^* \setminus \Gamma_u \to \Gamma_y \cap \Gamma_y^*}\mathbf{0}, \tag{19}$$

where $\mathbf{0}$ is a zero vector with appropriate dimensions. The underlying system is $(2r, 2s)$-sparse strongly observable, therefore we conclude that the sub-system $\hat{S} := (A, B_{(.,\Gamma_u \cup \Gamma_u^*)}, C_{(\Gamma_y \cap \Gamma_y^*,.)}, D_{(\Gamma_y \cap \Gamma_y^*, \Gamma_u \cup \Gamma_u^*)})$ is strongly observable. One can reinterpret both equations as two (possibly different) valid trajectories of the system $\hat{S}$ that share the same output sequence. Strong observability of $\hat{S}$ implies that $\hat{x} = x(t-n+1)$ which completes the proof.

The main algorithm in this paper builds upon this proposition. We search for a set of inputs and outputs that satisfies equality (17), i.e., we check if there exist $\hat{\mathbf{U}}$ and $\hat{x}$ that make equality (17) hold. Based on Proposition 11, we define a consistency check as follows,

Test 1 (*Consistency Check*)**.** *Given subsets of inputs and outputs denoted by Γ_u and Γ_y, TEST(Γ_u, Γ_y) returns true if:*

$$\min_{\hat{\mathbf{U}}, \hat{x}} \|\mathbf{Y}|_{\Gamma_y} - \mathscr{O}_{\Gamma_y}\hat{x} - \mathscr{N}_{\Gamma_u \to \Gamma_y}\hat{\mathbf{U}}\| \leq \varepsilon, \tag{20}$$

where $\varepsilon > 0$ is the solver tolerance, due to numerical errors. However, for the sake of clarity, we focus in this paper on the case when ε is negligible.[2]

Finding the right subset of inputs and outputs that satisfies this test is a combinatorial problem in nature and requires exhaustive search. It is well-known that secure state reconstruction under this attack model is in general *NP-hard* (Pasqualetti et al., 2013; Shoukry et al., 2017). This test is depicted in Algorithm 2.

In the rest of this section, we introduce an architecture for our observer followed by methods to improve its computational performance. For each input (output), we assign a Boolean variable $\mathbf{b}_i \in \mathbb{B}$ ($\mathbf{c}_i \in \mathbb{B}$) that indicates if the corresponding input (output) is under attack or not, i.e., $\mathbf{b}_i = 1$ ($\mathbf{c}_i = 1$) if the ith input (output) is under attack. In the rest of this paper, we use the bold letters ($\mathbf{b}$ and $\mathbf{c}$) to denote these Boolean variables and we reserve non-bold type face (b and c) as instances of them. Finding the right assignment of these Boolean variables is combinatorial in nature and in order to efficiently decide which set of inputs and outputs satisfies the TEST in (20), we design an observer using the lazy SMT paradigm (Barrett, Sebastiani, Seshia, & Tinelli, 2009).

[2] Note that the minimum always exists for (20) as the cost function is a semi-definite quadratic function.

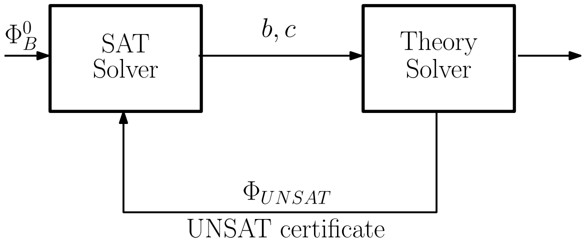

Fig. 2. The lazy SMT paradigm.

4.1. Overall architecture

The observer consists of two blocks that interact with each other, a propositional satisfiability (SAT) solver and a theory solver. A SAT solver is an algorithm that, given a propositional logic formula, finds an assignment to the truth values of the formula's atomic propositions so as to make the formula true. In this paper we consider solvers that can also handle pseudo-Boolean constraints/formulas, such as a counting the number of atomic propositions that are true or false. In the remainder of the paper we refer to atomic propositions as Boolean variables. The SAT solver will be used to produce an instance of $\mathbf{b} \in \mathbb{B}^m$ and $\mathbf{c} \in \mathbb{B}^p$ that will be checked for consistency, by the theory solver, using the consistency test. When the test fails, the inconsistency is encoded in a pseudo-Boolean constraint that is returned to the SAT solver. The general architecture is depicted in Fig. 2 and we now describe each of the steps in more detail.

The initial pseudo-Boolean constraint only bounds the number of attacked inputs and outputs, i.e.:

$$\Phi_B := (\sum_{i=1}^{m} \mathbf{b}_i \leq r) \bigwedge (\sum_{j=1}^{p} \mathbf{c}_i \leq s). \tag{21}$$

Initially, the SAT solver generates instances of $\mathbf{b}$ and $\mathbf{c}$ that satisfy Φ_B. The theory solver checks whether $\Gamma_u := \text{supp}(b)$ and $\Gamma_y := \overline{\text{supp}(c)}$ satisfies the consistency check. If the test is satisfied, then the algorithm terminates and returns the (delayed) reconstructed state. Otherwise, the theory solver outputs UNSAT and generates a reason for the conflict, a certificate, or a counterexample that is denoted by Φ_{cert}. This counterexample encodes the inconsistency among the chosen inputs and outputs. The following always constitutes a (naive) certificate:

$$\Phi_{\text{naive-cert}} := \sum_{i \in \text{supp}(b)} \mathbf{b}_i + \sum_{j \in \overline{\text{supp}(c)}} \mathbf{c}_j \geq 1. \tag{22}$$

On the next iteration, the SAT solver updates the constraint by conjoining Φ_{cert} to Φ_B, and generates another feasible assignment for $\mathbf{b}$ and $\mathbf{c}$. This procedure is repeated until the theory solver returns SAT as illustrated in Algorithm 1.

Algorithm 1. Secure state observer

```
Require: $A, B, C, D$ (system), $Y$ (output), $r, s$ (bounds)
status ← UNSAT
$\Phi_{\text{cert}} \leftarrow$ True
$\Phi_B \leftarrow (\sum_{i \in \{1,\dots,m\}} \mathbf{b}_i \leq r) \bigwedge (\sum_{i \in \{1,\dots,p\}} \mathbf{c}_i \leq s)$
while status = UNSAT do
    $\Phi_B \leftarrow \Phi_B \bigwedge \Phi_{\text{cert}}$
    $(b, c) \leftarrow$ SAT-solver($\Phi_B$)
    (status, x) ← T-solver.check(supp($b$), $\overline{\text{supp}(c)}$)
    $\Phi_{\text{cert}} \leftarrow$ T-solver.Certificate(supp($b$), $\overline{\text{supp}(c)}$)
return $(x, b, c)$
```

Note that Proposition 11 implies that the SAT solver eventually produces an assignment that satisfies the consistency test and therefore Algorithm 1 always terminates. The size of the certificate plays an important role in the overall execution time of the algorithm (Shoukry et al., 2017). Note that the attack model considered in Shoukry et al. (2017) is restricted to outputs, and the major contribution of our work is to handle both input and output attacks. In the next section, we focus on constructing shorter counterexamples to improve the run time.

Algorithm 2. T-solver.check

```
Require: $\Gamma_u, \Gamma_y$
Solve: $(\hat{x}, \hat{\mathbf{U}}) = \text{argmin}_{x,\mathbf{U}} \|\mathbf{Y}|_{\Gamma_y} - \mathscr{O}_{\Gamma_y} x - \mathscr{N}_{\Gamma_u \to \Gamma_y} \mathbf{U}\|$
if $\|\mathbf{Y}|_{\Gamma_y} - \mathscr{O}_{\Gamma_y} \hat{x} - \mathscr{N}_{\Gamma_u \to \Gamma_y} \hat{\mathbf{U}}\| \leq \varepsilon$ then
    status = SAT
else
    status = UNSAT
return (status, $\hat{x}$)
```

4.2. SAT certificate

In this part, we improve the efficiency of Algorithm 1 by constructing a shorter certificate (counter-example or conflicts). As it was discussed before, the naive certificate only excludes the current assignment of $\mathbf{b}$ and $\mathbf{c}$ from the search space of the SAT solver, however, by exploiting the structure of the underlying system, we show that we can further decrease the size of the certificate and therefore prune the search space more efficiently.

One of the main results of this paper is to show that we can always find a smaller conflicting subset of inputs and outputs. We propose two methods for generating shorter certificates. The first method reduces the size of the counterexample by at least $s - 1$, we explain this method in Lemma 13 and give a formal proof of the existence of such shorter certificate. In practice, however we observe the reduction in the length of conflicts is much larger than this theoretical bound. The second method is inspired by the QuickXplain algorithm. This method generates counter-examples that are irreducible, meaning that we cannot reduce the size of the counter-example by removing some of it's entries. We also note that by generating multiple certificates at each iteration we can further enhance the execution time. At the end of this section Lemma 15 states that for a generic LTI system the size of the certificate cannot be smaller than $m + 1$.

Let us assume the SAT solver hypothesized $\Gamma_u^{\text{SAT}} := \text{supp}(b)$ and $\Gamma_y^{\text{SAT}} := \overline{\text{supp}(c)}$ as the set of compromised inputs and safe outputs, respectively. The main intuition behind both methods is to look for $\Gamma_u^{\text{cert}} \supseteq \Gamma_u^{\text{SAT}}$ and $\Gamma_y^{\text{cert}} \subseteq \Gamma_y^{\text{SAT}}$ that would not satisfy the consistency test. Note that the certificate consists of inputs in $\overline{\Gamma}_u^{\text{cert}}$ and outputs in Γ_y^{cert}.

4.3. Method I based on heuristics

Method I reduces the size of the certificate by increasing the size of (supposedly under attack) inputs (Γ_u^{cert}) followed by decreasing the size of (supposedly safe) outputs (Γ_u^{cert}). The summary of the above procedure of shortening certificates is illustrated in Algorithm 3. We begin by adding inputs to Γ_u^{SAT} while making sure TEST still returns false and the number of inputs is bounded by $2r$. Let us denote this new set of inputs by Γ_u^{cert}.

At the second step, we shrink the set of conflicting outputs in order to further shorten the size of the counterexample. Let us denote a subset of Γ_y^{SAT} of size $p - 2s$ by Γ_y^{temp}. The following lemma shows we can reduce the size of conflicting outputs at least by $s - 1$.

Algorithm 3. T-solver.Certificate 1

```
Require: $\Gamma_u^{\mathrm{SAT}}, \Gamma_y^{\mathrm{SAT}}$
    step 1: Conduct a linear search in the input set
Sort $\overline{\Gamma}_u^{\mathrm{SAT}}$
status ← UNSAT, $j \leftarrow \emptyset$, $\Gamma_u^{\mathrm{cert}} \leftarrow \Gamma_u^{\mathrm{SAT}}$
while status == UNSAT and $|\Gamma_u^{\mathrm{cert}}| < 2r$ do
    $\Gamma_u^{\mathrm{cert}} \leftarrow \Gamma_u^{\mathrm{cert}} \cup \{j\}$
    pick another input $j \in \overline{\Gamma}_u^{\mathrm{SAT}}$
    (status, $x$) ← T-Solver.check($\Gamma_u^{\mathrm{cert}} \cup \{j\}$, $\Gamma_y^{\mathrm{SAT}}$)
    step 2: Conduct a linear search in the output set
Sort $\Gamma_y^{\mathrm{SAT}}$
Pick a subset of size $p - 2s$: $\Gamma_y^{\mathrm{temp}} \subseteq \Gamma_y^{\mathrm{SAT}}$
status ← SAT, $i \leftarrow \emptyset$
while status == SAT do
    $\Gamma_y^{\mathrm{cert}} \leftarrow \Gamma_y^{\mathrm{temp}} \cup \{i\}$
    (status, $x$) ← T-Solver.check($\Gamma_u^{\mathrm{cert}}$, $\Gamma_y^{\mathrm{cert}}$)
    Pick another output $i \in \Gamma_y^{\mathrm{SAT}} \setminus \Gamma_y^{\mathrm{temp}}$
$\Phi_{\mathrm{cert}}^1 \leftarrow \sum_{j \in \overline{\Gamma}_u^{\mathrm{cert}}} \mathbf{b}_j + \sum_{i \in \Gamma_y^{\mathrm{cert}}} \mathbf{c}_i \geq 1$
return $\Phi_{\mathrm{cert}}^1$
```

Lemma 13. *Consider an LTI system, given by (1), assume it is $(2r, 2s)$-sparse strongly observable, and that the number of attacked inputs and outputs is bounded by r and s, respectively. Pick any subset of inputs and outputs denoted by Γ_u^{cert} and Γ_y^{SAT} with $|\Gamma_u^{cert}| \leq 2r$ and $|\Gamma_y^{SAT}| \geq p - s$, that do not satisfy the consistency check (20). Given any subset of at most $p-2s$ outputs denoted by $\Gamma_y^{temp} \subseteq \Gamma_y^{SAT}$, one of the following is true:*

1. *TEST(Γ_u^{cert}, Γ_y^{temp}) returns false,*
2. *There exists an output $i \in \Gamma_y^{SAT} \setminus \Gamma_y^{temp}$ such that TEST(Γ_u^{cert}, $\Gamma_y^{temp} \cup \{i\}$) returns false.*

Proof. Please refer to Appendix.

We denote this smaller set of conflicting outputs Γ_y^{temp} (if TEST($\Gamma_u^{\mathrm{cert}}, \Gamma_y^{\mathrm{temp}}$) returns false, otherwise $\Gamma_y^{\mathrm{temp}} \cup \{i\}$) by Γ_y^{cert}. Lemma 13 gives formal guarantees of the existence of shorter certificates which hold no matter how the subsets of inputs and outputs (Γ_u^{temp} and Γ_y^{temp}) are chosen. This lemma shows that Method I reduces the size of the certificate by at least $s - 1$.

In practice, we choose these subsets based on heuristics that have for objective a decrease in the overall running time. We assign slack variables to inputs and outputs similarly to Shoukry et al. (2017) and Showkatbakhsh et al. (2017), and sort them based on the structure of the system. Recall that Algorithm 3 shortens the certificate by reducing the number of inputs followed by the reduction in the number of outputs, i.e., we *simultaneously* reducing both inputs and outputs in the certificate. We observe that by generating two counterexamples, we can prune the search space of the SAT solver more efficiently. Similarly to Algorithm 5, we can find two counterexamples by reducing the number of inputs following a reduction in the number of outputs and vice-verse.

Sorting $\overline{\Gamma}_u^{\mathrm{SAT}}$ and Γ_y^{SAT}:

Assuming TEST($\overline{\Gamma}_u^{\mathrm{SAT}}, \Gamma_y^{\mathrm{SAT}}$) returns false, we assign slack variables to inputs in $\overline{\Gamma}_u^{\mathrm{SAT}}$ and outputs in Γ_y^{SAT}, denoted by $\mathrm{slack}_u(j)$ and $\mathrm{slack}_y(i)$, respectively. Let us denote a solution to the optimization (20) inside TEST($\Gamma_u^{\mathrm{SAT}}, \Gamma_y^{\mathrm{SAT}}$) by $\hat{x}$ and $\hat{\mathbf{U}}$.

We define $\mathrm{slack}_u(j)$ for $j \in \overline{\Gamma}_u^{\mathrm{SAT}}$ as the norm of the projection of $\mathbf{Y}|_{\Gamma_y^{\mathrm{SAT}}} - \mathscr{O}_{\Gamma_y^{\mathrm{SAT}}}\hat{x} - \mathscr{N}_{\Gamma_u^{\mathrm{SAT}} \to \Gamma_y^{\mathrm{SAT}}}\hat{\mathbf{U}}$ onto the column space of $\mathscr{N}_{j \to \Gamma_y^{\mathrm{SAT}}}$:

$$\mathrm{slack}_u(j) := \tag{23}$$

$$\| \mathscr{N}_{j \to \Gamma_y^{\mathrm{SAT}}} \mathscr{N}^{\dagger}_{j \to \Gamma_y^{\mathrm{SAT}}} \left(\mathbf{Y}|_{\Gamma_y^{\mathrm{SAT}}} - \mathscr{O}_{\Gamma_y^{\mathrm{SAT}}}\hat{x} - \mathscr{N}_{\Gamma_u^{\mathrm{SAT}} \to \Gamma_y^{\mathrm{SAT}}}\hat{\mathbf{U}} \right) \| .$$

This slack variable measures how much of the residual can be justified by considering j in addition to Γ_u^{SAT}. Note that we want to append inputs to Γ_u^{SAT} while having a false TEST. We first normalize these slack variables by the norm of the corresponding invertibility matrix, and $\overline{\Gamma}_u^{\mathrm{SAT}}$ is obtained by sorting slack variables in *ascending* order.

We define $\mathrm{slack}_y(i)$ as the residual of each output:

$$\mathrm{slack}_y(i) := \|\mathbf{Y}|_i - \mathscr{O}_i\hat{x} - \mathscr{N}_{\Gamma_u^{\mathrm{SAT}} \to \{i\}}\mathbf{U}\|, \quad i \in \Gamma_y^{\mathrm{SAT}}. \tag{24}$$

Note that:

$$\sum_{i \in \Gamma_u^{\mathrm{SAT}}} \mathrm{slack}_y(i) = \min_{\hat{\mathbf{U}}, \hat{x}} \|\mathbf{Y}|_{\Gamma_y^{\mathrm{SAT}}} - \mathscr{O}_{\Gamma_y^{\mathrm{SAT}}}\hat{x} - \mathscr{N}_{\Gamma_u^{\mathrm{SAT}} \to \Gamma_y^{\mathrm{SAT}}}\hat{\mathbf{U}}\|. \tag{25}$$

We first normalize each slack variable by the norm of the corresponding observability matrix. Recall that we aim to find a smaller subset of Γ_u^{SAT} while ensuring TEST returns false. We pick the output with the highest slack variable as the first element of Γ_u^{SAT}. We sort the rest based on the dimension of the kernel of each observability matrix, following the intuition provided in Shoukry et al. (2017).

4.4. Method II based on QuickXplain

The second method (Algorithm 5) is inspired by QuickXplain and generates a counter-example by pruning the naive-certificate (22) to make it irreducible. We formally define this property as follows.

Definition 14 (*Irreducible Certificate*)**.** A certificate consisting of inputs $\overline{\Gamma}_u$ and outputs Γ_y is irreducible, if no other subset of it can generate a conflict, i.e., for all subsets denoted by $\overline{\Gamma}'_u \subseteq \overline{\Gamma}_u$ and $\Gamma'_y \subseteq \Gamma_y$ the following are equivalent:

1. $\overline{\Gamma}'_u$ and Γ'_y generate a conflict.
2. $\overline{\Gamma}'_u = \overline{\Gamma}_u$ and $\Gamma'_y = \Gamma_y$.

One cannot prune irreducible certificates and each element is necessary for the set to remain a counter-example. Let Δ^{SAT} be the elements (consisting of inputs $\overline{\Gamma}_u^{\mathrm{SAT}}$ and outputs Γ_y^{SAT}) of the naive certificate. For ease of exposition we slightly abuse notation to denote TEST($\Gamma_u^{\mathrm{SAT}}, \Gamma_y^{\mathrm{SAT}}$) by TEST($\Delta^{\mathrm{SAT}}$). We denote the output of this algorithm by Δ_{cert} which consists of inputs $\overline{\Gamma}_u^{\mathrm{cert}}$ and outputs Γ_y^{cert}.

This method consists of an exploration phase in which it finds an element (input or output) that belongs to an irreducible certificate. Let us denote an enumeration of Δ^{SAT} by $e_1, \dots, e_k$, and the internal state by $\Delta_{\mathrm{temp}} \leftarrow \emptyset$. This method begins by adding step-by-step elements of Δ^{SAT} to Δ_{temp}. The first element ($e_i \in \Delta^{\mathrm{SAT}}$) that fails TEST($\Delta_{\mathrm{temp}}$) is part of an irreducible certificate, and therefore is added to Δ_{cert}.

In order to find further elements of this certificate, we keep e_i in the background and the first element that fails the consistency check is added to Δ_{cert}. This repeated process can be implemented efficiently by using the divide and conquer paradigm as depicted in Algorithm 4. When an element e_i of Δ^{SAT} is detected we divide the remaining elements into two disjoint subsets $\Delta^1 := \{e_1, \dots, e_j\}$ and $\Delta^2 := \{e_{j+1}, \dots, e_{i-1}\}$. We can now recursively apply the algorithm to find a conflict Δ^2_{cert} among Δ^2 by keeping the set Δ^1 in the background and a conflict Δ^1_{cert} among Δ^1 by keeping the set Δ^2_{cert} in the background. This method of finding an irreducible subset is depicted in Algorithm 4

Note that the resulting counter-example depends on the initial enumeration of elements in Δ^{SAT}. If all the inputs (outputs) are ahead of outputs (inputs), then the resulting counter-example mostly consists of inputs (outputs). In order to have the maximal reduction in the search space of the SAT solver at each iteration, we produce three certificate using this method, putting inputs first, outputs first and mixing both inputs and outputs.

In the last part of this section, we look at the certificate size for a generic LTI system. We observe that the certificate size cannot be smaller that the number of inputs which is stated formally in the following lemma.

Algorithm 4. T-solver.QuickXplain

```
Require: $\Delta^0_{\text{cert}}, \Delta^0$
if T-solver.check($\Delta^0_{\text{cert}}$) = UNSAT or $\Delta^0 == \emptyset$ then
    return $\emptyset$
Let $e_1, \cdots, e_k$ be an enumeration of $\Delta^0$
$i \leftarrow 0$, $\Delta_{\text{temp}} \leftarrow \Delta^0_{\text{cert}}$,
while T-solver.check($\Delta_{\text{temp}}$) = SAT and $i \leq k$ do
    $i \leftarrow i + 1$
    $\Delta_{\text{temp}} \leftarrow \Delta_{\text{temp}} \cup e_i$
    $\Delta^i_{\text{cert}} \leftarrow \Delta_{\text{temp}}$
$\Delta_{\text{cert}} \leftarrow e_i, j \leftarrow \lfloor \frac{i}{2} \rfloor$
$\Delta^1 \leftarrow \{e_1, \cdots, e_j\}$
$\Delta^2 \leftarrow \{e_{j+1}, \cdots, e_{i-1}\}$
$\Delta_{\text{cert}} \leftarrow \Delta_{\text{cert}} \cup$ T-solver.QuickXplain($\Delta^j_{\text{cert}} \cup \Delta_{\text{cert}}, \Delta^2$)
$\Delta_{\text{cert}} \leftarrow \Delta_{\text{cert}} \cup$ T-solver.QuickXplain($\Delta^0_{\text{cert}} \cup \Delta_{\text{cert}}, \Delta^1$)
return $\Delta_{\text{cert}}$
```

Algorithm 5. T-solver.Certificate 2

```
Require: $\Gamma^{\text{SAT}}_u, \Gamma^{\text{SAT}}_y$
$\Delta_{\text{cert}} \leftarrow$ T-solver.QuickXplain($\emptyset, \overline{\Gamma}^{\text{SAT}}_u \cup \Gamma^{\text{SAT}}_y$)
Divide $\Delta_{\text{cert}}$ to inputs $\overline{\Gamma}^{\text{cert}}_u$ and outputs $\Gamma^{\text{cert}}_y$
$\Phi^2_{\text{cert}} \leftarrow \sum_{j \in \overline{\Gamma}^{\text{cert}}_u} \mathbf{b}_j + \sum_{i \in \Gamma^{\text{cert}}_y} \mathbf{c}_i \geq 1$
return $\Phi^2_{\text{cert}}$
```

Lemma 15. *For an LTI system, given by (1), the size of the certificate is always lower bounded by $m+1$, where m is the number of inputs.*

Proof. Please refer to Appendix.

5. Simulation results

We implemented our SMT-based observer in MATLAB while interfacing with the SAT solver SAT4J (Le Berre & Parrain, 2010) and assessed its performance in two case studies, randomly generated LTI systems and a chemical plant. We report the overall running time by using the two proposed methods, Algorithms 3 and 5.

5.1. Random systems

We randomly generate systems with a fixed state dimension ($n = 40$) and increase the number of inputs and outputs. Each system is generated by drawing entries of (A, B, C, D) according to uniform distribution, when necessary we scale A to ensure that the spectral radius is close to one. In each experiment, twenty percent of inputs and outputs are under adversarial attacks, and we generate the support set for the adversarial signals uniformly at random. Attack signals and the initial states are drawn according to independent and normally distributed random variables with zero mean and unit variance. All the systems under experiment satisfy a suitable sparse strong observability condition as described in Section 3.

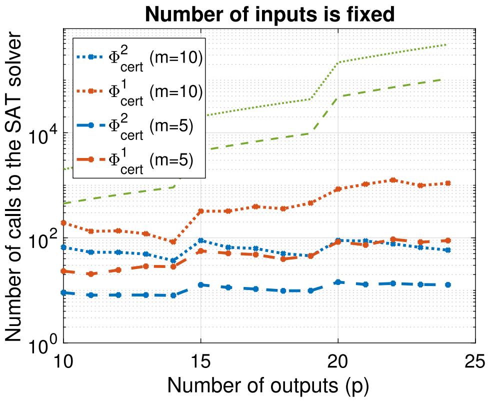

Fig. 3. Number of calls to the SAT solver in Algorithm 1 using Φ^1_{cert}, Φ^2_{cert} versus the number of outputs (p) for a fixed number of inputs. Green dotted and green dashed lines are upper-bounds for the number of the SAT solver calls when using the naive certificate for $m = 5$ and $m = 10$, respectively.

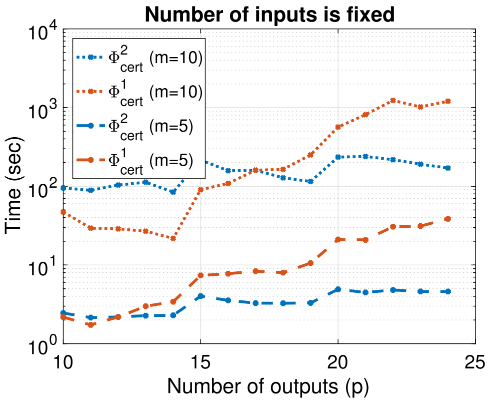

Fig. 4. Execution time of Algorithm 1 using Φ^1_{cert}, Φ^2_{cert} versus the number of outputs (p) and inputs (m).

Figs. 3 and 4 report the results of the simulations, each point represents the average of 20 experiments. All the experiments run on an Intel Core i5 2.7 GHz processor with 16 GB of RAM. We verify the run-time improvement resulting from using the shorter certificates, Φ^1_{cert} and Φ^2_{cert}, compared to the theoretical upper-bound of the brute-force approach in Fig. 3. For instance, consider the scenario with $p = 24$ and $m = 10$ in Figs. 3 and 4. In the brute-force approach, we require to check all $\binom{24}{4} \times \binom{10}{2} \approx 4.8 \times 10^5$ different combinations of inputs and outputs, however, by exploiting either Φ^1_{cert} or Φ^2_{cert} we observe a substantial improvement. We observe that although Φ^2_{cert} gives a worse run time for systems with smaller number of outputs, it scales better compared to Φ^1_{cert} when the number of inputs and outputs grow.

5.2. Chemical plant

In this part, we use the proposed observer to detect attacks on inputs and outputs of a simplified version of the Tennessee Eastman control challenge problem (Downs & Vogel, 1993). Ricker (1993) derived a continuous time LTI model of the plant interaction in its steady state. This system consists of 4 control inputs and 10 measured outputs and the linearized model has 8

Table 1
Average performance of the proposed observer.

	Overall execution time	Number of calls to the SAT solver
Φ^1_{cert}	0.22 s	20.05
Φ^2_{cert}	0.21 s	7.95

state variables. The structure of the continuous-time dynamics is reported below.

$$\frac{dx}{dt} = \begin{bmatrix} \ast & \ast & \ast & \ast & \ast & \ast & \ast & 0 \\ \ast & \ast & \ast & \ast & \ast & 0 & \ast & 0 \\ \ast & \ast & \ast & \ast & \ast & 0 & \ast & 0 \\ \ast & \ast & \ast & \ast & 0 & 0 & 0 & \ast \\ 0 & 0 & 0 & 0 & \ast & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \ast & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ast & 0 \\ 0 & 0 & 0 & \ast & 0 & 0 & 0 & \ast \end{bmatrix} x + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \ast & 0 & 0 & 0 \\ 0 & \ast & 0 & 0 \\ 0 & 0 & \ast & 0 \\ 0 & 0 & 0 & \ast \end{bmatrix} u,$$

$$y = \begin{bmatrix} 0 & 0 & 0 & 0 & \ast & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \ast & 0 & 0 \\ \ast & \ast & \ast & \ast & 0 & 0 & \ast & 0 \\ \ast & \ast & \ast & \ast & 0 & 0 & 0 & \ast \\ \ast & \ast & \ast & \ast & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \ast & 0 & 0 & 0 & 0 \\ \ast & \ast & \ast & 0 & 0 & 0 & 0 & 0 \\ \ast & \ast & \ast & 0 & 0 & 0 & 0 & 0 \\ \ast & \ast & \ast & 0 & 0 & 0 & 0 & 0 \\ \ast & \ast & \ast & 0 & 0 & 0 & \ast & \ast \end{bmatrix} x,$$

where $\ast$ represents a non-zero entry,[3] and $x \in \mathbb{R}^8$, $u \in \mathbb{R}^4$ and $y \in \mathbb{R}^{10}$ are state, input and output variables, respectively. The only known limitation of this LTI model is the system should operate close to its steady-state. We obtain a discrete-time model by discretizing the continuous-time model assuming a zero-order hold for the input u, with a time-step of 5s. The attacker can read all the inputs and outputs and manipulate one control input and two measured outputs. The discrete-time model is $(2, 4)$-sparse strongly observable, therefore our observer can correctly reconstruct the state under this attack model.

We randomly generate attack signals and the initial state according to independent and normally distributed random variables. The support set of attacks are drawn uniformly at random, and in each experiment one input and two outputs are under adversarial attacks.

The proposed observer in this paper can correctly reconstruct the (delayed) state after 8 samples, and the average performance of 20 experiments, by using Φ^1_{cert} and Φ^2_{cert} is reported in Table 1. The overall execution time is the run time of the observer after receiving all the required samples from the plant, and it does not take the sampling time of the plant into account. We observe that the execution time of the observer to reconstruct the state and to detect attacks is much smaller compared to the sampling time of the plant.

6. Conclusion

In this paper, we considered the problem of secure state reconstruction when inputs and/or outputs are under adversarial attacks. In this set-up, there is no restriction on how the adversary manipulates inputs and outputs. By introducing the notion of sparse strong observability, we derived necessary and sufficient conditions under which state reconstruction is possible given bounds on the number of attacked outputs and inputs. Furthermore, we demonstrated the scalability and effectiveness of the proposed observer with numerical simulations.

Appendix

Proof of Lemma 5. We first prove the sufficiency part. For the sake of contradiction, suppose that the underlying system is not strongly observable but the property of Lemma 5 is true. If the underlying system (6) is not strongly observable, it means there exist two initial conditions, denoted by $x^{(1)}(0)$ and $x^{(2)}(0)$ possibly with different input sequences denoted by $\{u^{(1)}(t)\}$ and $\{u^{(2)}(t)\}$, respectively, that correspond to the same output sequence $\{y(t)\}$. The underlying system is linear, therefore the nonzero initial condition of $x^{(1)}(0) - x^{(2)}(0)$ with the input sequence $\{u^{(1)}(t) - u^{(2)}(t)\}$ produces the zero output sequence which contradicts the property given in Lemma 5. The necessity can be concluded using the similar argument. For the sake of contradiction let us assume this property does not hold, i.e., there exists a non zero initial state $x(0) \neq 0$ that corresponds to the zero output sequence. This contradicts the strong observability since the zero output sequence can be generated from both zero and $x(0) \neq 0$ as initial conditions under (possibly different) input sequences.

Proof of Lemma 13. We prove this lemma with contradiction. We show that if TEST$(\Gamma_u^{\text{cert}}, \Gamma_y^{\text{temp}} \cup \{i\})$ returns true for all $i \in \Gamma_y^{\text{SAT}} \setminus \Gamma_y^{\text{temp}}$ then TEST$(\Gamma_u^{\text{cert}}, \Gamma_y^{\text{SAT}})$ would also return true, which contradicts the assumption of the lemma. By applying the following lemma successively, the result follows directly.

Lemma 16. *Consider an LTI system, given by (1), and assume it is $(2r, 2s)$-sparse strongly observable. Pick any subset of inputs and outputs denoted by Γ_u^{cert} and Γ_y^{temp} with $|\Gamma_u^{cert}| \le 2r$ and $|\Gamma_y^{temp}| \ge p - 2s$. Then for any subsets of outputs denoted by Γ_y^1 and Γ_y^2, the first statement implies the second:*

(1) TEST$(\Gamma_u^{cert}, \Gamma_y^{temp} \cup \Gamma_y^1)$ and TEST$(\Gamma_u^{cert}, \Gamma_y^{temp} \cup \Gamma_y^2)$ return true.
(2) TEST$(\Gamma_u^{cert}, \Gamma_y^{temp} \cup \Gamma_y^1 \cup \Gamma_y^2)$ returns true.

Proof. Without loss and generality we can assume Γ_y^1, Γ_y^2 and Γ_y^{temp} are all disjoint sets. Since TEST$(\Gamma_u^{\text{cert}}, \Gamma_y^{\text{temp}} \cup \Gamma_y^i)$ returns true for $i \in \{1, 2\}$, therefore we have:

$$\begin{bmatrix} \mathbf{Y}|_{\Gamma_y^{\text{temp}}} \\ \mathbf{Y}|_{\Gamma_y^1} \end{bmatrix} = \begin{bmatrix} \mathscr{O}_{\Gamma_y^{\text{temp}}} \\ \mathscr{O}_{\Gamma_y^1} \end{bmatrix} \hat{x}^1 + \begin{bmatrix} \mathscr{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^{\text{temp}}} \\ \mathscr{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^1} \end{bmatrix} \hat{\mathbf{U}}^1, \tag{A.1}$$

$$\begin{bmatrix} \mathbf{Y}|_{\Gamma_y^{\text{temp}}} \\ \mathbf{Y}|_{\Gamma_y^2} \end{bmatrix} = \begin{bmatrix} \mathscr{O}_{\Gamma_y^{\text{temp}}} \\ \mathscr{O}_{\Gamma_y^2} \end{bmatrix} \hat{x}^2 + \begin{bmatrix} \mathscr{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^{\text{temp}}} \\ \mathscr{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^2} \end{bmatrix} \hat{\mathbf{U}}^2, \tag{A.2}$$

where $\hat{x}^1, \hat{x}^2 \in \mathbb{R}^n$ are states that T-solver.check returns, $\hat{\mathbf{U}}^1, \hat{\mathbf{U}}^2$ are matrices with appropriate dimensions that satisfy TEST. Note that the underlying system is $(2r, 2s)$-sparse strongly observable, $|\Gamma_u^{\text{cert}}| \le 2r$ and $|\Gamma_y^{\text{temp}}| \ge p - s$ therefore $\hat{S} := (A, B_{(.,\Gamma_u^{\text{cert}})}, C_{(\Gamma_y^{\text{temp}},.)}, D_{(\Gamma_y^{\text{temp}},\Gamma_u^{\text{cert}})})$ is strongly observable. One can reinterpret $(\hat{\mathbf{U}}^1, \mathbf{Y}|_{\Gamma_y^{\text{temp}}})$ and $(\hat{\mathbf{U}}^2, \mathbf{Y}|_{\Gamma_y^{\text{temp}}})$ as two (possibly different) valid trajectories of a strongly observable system $\hat{S}$ with identical output sequences. Strong observability implies that the state can be uniquely determined from the output with a delay bounded by n, therefore $\hat{x}^1 = \hat{x}^2$. Furthermore, the equality of right hand sides of (A.1) and (A.2) implies that:

$$\mathscr{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^{\text{temp}}} (\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1) = 0, \tag{A.3}$$

i.e., $\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1$ is a zero dynamic of $\hat{S}$. By $(2r, 2s)$-sparse strongly observability of S, we conclude that $\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1$ is also a zero dynamic of S, and therefore,

$$\mathscr{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^1} (\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1) = 0, \quad \mathscr{N}_{\Gamma_u^{\text{cert}} \to \Gamma_y^2} (\hat{\mathbf{U}}^2 - \hat{\mathbf{U}}^1) = 0. \tag{A.4}$$

[3] For the exact dynamics of the LTI model, see Ricker (1993).

Putting (A.1), (A.2) and (A.4) together with $\hat{x}^1 = \hat{x}^2$, we conclude that:

$$\begin{bmatrix} \mathbf{Y}|_{\varGamma_y^{\text{temp}}} \\ \mathbf{Y}|_{\varGamma_y^1} \\ \mathbf{Y}|_{\varGamma_y^2} \end{bmatrix} = \begin{bmatrix} \mathscr{O}_{\varGamma_y^{\text{temp}}} \\ \mathscr{O}_{\varGamma_y^1} \\ \mathscr{O}_{\varGamma_y^2} \end{bmatrix} \hat{x}^1 + \begin{bmatrix} \mathscr{N}_{\varGamma_u^{\text{cert}} \to \varGamma_y^{\text{temp}}} \\ \mathscr{N}_{\varGamma_u^{\text{cert}} \to \varGamma_y^1} \\ \mathscr{N}_{\varGamma_u^{\text{cert}} \to \varGamma_y^2} \end{bmatrix} \hat{\mathbf{U}}^1, \tag{A.5}$$

i.e., TEST($\varGamma_u^{\text{cert}}, \varGamma_y^{\text{temp}} \cup \varGamma_y^1 \cup \varGamma_y^2$) returns false.

Proof Sketch of Lemma 15. Let us revisit the optimization (20) inside the consistency check TEST($\varGamma_u, \varGamma_y$):

$$\min_{\hat{x},\hat{\mathbf{U}}} \left\| \mathbf{Y}|_{\varGamma_y} - \left[\mathscr{O}_{\varGamma_y}, \mathscr{N}_{\varGamma_u \to \varGamma_y}\right] \begin{bmatrix} \hat{x} \\ \hat{\mathbf{U}} \end{bmatrix} \right\|. \tag{A.6}$$

For a generic LTI system, the matrix $\left[\mathscr{O}_{\varGamma_y}, \mathscr{N}_{\varGamma_u \to \varGamma_y}\right] \in \mathbb{R}^{n|\varGamma_y| \times n(1+|\varGamma_u|)}$ is of full rank, where n is the order of the LTI system. If $\left[\mathscr{O}_{\varGamma_y}, \mathscr{N}_{\varGamma_u \to \varGamma_y}\right] \in \mathbb{R}^{n|\varGamma_y| \times n(1+|\varGamma_u|)}$ is of full row rank, then TEST($\varGamma_u, \varGamma_y$) is satisfied irrespectively of the actual values of $\mathbf{Y}|_{\varGamma_y}$. Therefore in order to have a certificate constructed by inputs in $\overline{\varGamma}_u$ and outputs in $\varGamma_y$, $\left[\mathscr{O}_{\varGamma_y}, \mathscr{N}_{\varGamma_u \to \varGamma_y}\right] \in \mathbb{R}^{n|\varGamma_y| \times n(1+|\varGamma_u|)}$ should be a full column rank matrix, therefore:

$$n|\varGamma_y| \geq n(1 + |\varGamma_u|). \tag{A.7}$$

The certificate consists of inputs in $\overline{\varGamma}_u$ and outputs in $\varGamma_y$, therefore the length of certificate is:

$$|\overline{\varGamma}_u| + |\varGamma_y| = m - |\varGamma_u| + |\varGamma_y| \geq m + 1. \tag{A.8}$$

References

Amin, Saurabh, Schwartz, Galina A., & Hussain, Amir (2013). In quest of benchmarking security risks to cyber-physical systems. *IEEE Network*, *27*(1), 19–24.

Bai, Cheng-Zong, Gupta, Vijay, & Pasqualetti, Fabio (2017). On kalman filtering with compromised sensors: Attack stealthiness and performance bounds. *IEEE Trans. Automat. Control*, *62*(12), 6641–6648.

Bai, Cheng-Zong, Pasqualetti, Fabio, & Gupta, Vijay (2017). Data-injection attacks in stochastic control systems: Detectability and performance tradeoffs. *Automatica*, *82*, 251–260.

Barrett, Clark W., Sebastiani, Roberto, Seshia, Sanjit A., & Tinelli, Cesare (2009). Satisfiability modulo theories. *Handbook of Satisfiability*, *185*, 825–885.

Cárdenas, Alvaro A., Amin, Saurabh, & Sastry, Shankar (2008). Research challenges for the security of control systems. In *Conference on hot topics in security (hotsec)*.

Chong, Michelle S., Wakaiki, Masashi, & Hespanha, Joao P. (2015). Observability of linear systems under adversarial attacks. In *American control conference (ACC)* (pp. 2439–2444).

De Persis, Claudio, & Tesi, Pietro (2015). Input-to-state stabilizing control under denial-of-service. *IEEE Trans. Automat. Control*, *60*(11), 2930–2944.

Downs, James J., & Vogel, Ernest F. (1993). A plant-wide industrial process control problem. *Computers & Chemical Engineering*, *17*(3), 245–255.

Fawzi, Hamza, Tabuada, Paulo, & Diggavi, Suhas (2014). Secure estimation and control for cyber-physical systems under adversarial attacks. *IEEE Trans. Automat. Control*, *59*(6), 1454–1467.

Giraldo, Jairo, Urbina, David, Cardenas, Alvaro, Valente, Junia, Faisal, Mustafa, Ruths, Justin, et al. (2018). A survey of physics-based attack detection in cyber-physical systems. *ACM Comput. Surv.*, *51*(4), 76.

Greenberg, Andy (2015). Hackers remotely kill a jeep on the highway, with me in it. [online] http://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway.

Gupta, Abhishek, Langbort, Cédric, & Basar, Tamer (2010). Optimal control in the presence of an intelligent jammer with limited actions. In *49th IEEE conference on decision and control (CDC)* (pp. 1096–1101).

Harirchi, Farshad, & Ozay, Necmiye (2016). Guaranteed model-based fault detection in cyber-physical systems: A model invalidation approach. arXiv preprint arXiv:1609.05921.

Hautus, M. L. J. (1983). Strong detectability and observers. *Linear Algebra Appl.*, *50*(Supplement C), 353–368.

Junker, Ulrich (2001). Quickxplain: Conflict detection for arbitrary constraint propagation algorithms. In *IJCAI'01 workshop on modelling and solving problems with constraints*.

Kelion, Leo (2016). Nissan leaf electric cars hack vulnerability disclosed. [online] http://www.bbc.com/news/technology-35642749.

Langner, Ralph (2011). Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security & Privacy*, *9*(3), 49–51.

Le Berre, Daniel, & Parrain, Anne (2010). The sat4j library, release 2.2, system description. *J. Satisf. Boolean Model. Comput.*, *7*, 59–64.

Mishra, Shaunak, Shoukry, Yasser, Karamchandani, Nikhil, Diggavi, Suhas, & Tabuada, Paulo (2017). Secure state estimation: Optimal guarantees against sensor attacks in the presence of noise. *IEEE Trans. Control Netw. Syst.*, *4*(1), 49–59.

Mo, Yilin, Chabukswar, Rohan, & Sinopoli, Bruno (2014). Detecting integrity attacks on scada systems. *IEEE Trans. Control Syst. Technol.*, *22*(4), 1396–1407.

Mo, Yilin, Garone, Emanuele, Casavola, Alessandro, & Sinopoli, Bruno (2010). False data injection attacks against state estimation in wireless sensor networks. In *49th IEEE conference on decision and control (CDC)* (pp. 5967–5972).

Mo, Yilin, Kim, Tiffany Hyun-Jin, Brancik, Kenneth, Dickinson, Dona, Lee, Heejo, Perrig, Adrian, et al. (2012). Cyber–physical security of a smart grid infrastructure. *Proc. IEEE*, *100*(1), 195–209.

Mo, Yilin, & Sinopoli, Bruno (2009). Secure control against replay attacks. In *Allerton conference on communication, control, and computing* (pp. 911–918).

Mo, Yilin, & Sinopoli, Bruno (2016). On the performance degradation of cyber-physical systems under stealthy integrity attacks. *IEEE Trans. Automat. Control*, *61*(9), 2618–2624.

Nakahira, Yorie, & Mo, Yilin (2015). Dynamic state estimation in the presence of compromised sensory data. In *54th annual conference on decision and control (CDC)* (pp. 5808–5813).

Pajic, Miroslav, Weimer, James, Bezzo, Nicola, Tabuada, Paulo, Sokolsky, Oleg, Lee, Insup, et al. (2014). Robustness of attack-resilient state estimators. In *ICCPS'14: ACM/IEEE 5th international conference on cyber-physical systems (with CPS Week 2014)* (pp. 163–174).

Pasqualetti, Fabio, Dorfler, Florian, & Bullo, Francesco (2013). Attack detection and identification in cyber-physical systems. *IEEE Trans. Automat. Control*, *58*(11), 2715–2729.

Ricker, Lawrence (1993). Model predictive control of a continuous, nonlinear, two-phase reactor. *J. Process Control*, *3*(2), 109–123.

Sandberg, Henrik, & Teixeira, André M. H. (2016). From control system security indices to attack identifiability. In *Science of security for cyber-physical systems workshop (SOSCYPS)* (pp. 1–6).

Senejohnny, Danial, Tesi, Pietro, & De Persis, Claudio (2016). A jamming-resilient algorithm for self-triggered network coordination. arXiv preprint arXiv:1603.02563.

Shoukry, Yasser, Nuzzo, Pierluigi, Puggelli, Alberto, Sangiovanni-Vincentelli, Alberto L., Seshia, Sanjit A., & Tabuada, Paulo (2017). Secure state estimation for cyber physical systems under sensor attacks: a satisfiability modulo theory approach. *IEEE Trans. Automat. Control*, *62*(10), 4917–4932.

Shoukry, Yasser, & Tabuada, Paulo (2016). Event-triggered state observers for sparse sensor noise/attacks. *IEEE Trans. Automat. Control*, *61*(8), 2079–2091.

Showkatbakhsh, Mehrdad, Shoukry, Yasser, Chen, Robert, Diggavi, Suhas, & Tabuada, Paulo (2017). An SMT-based approach to secure state estimation under sensor and actuator attacks. In *IEEe conference on decision and control (CDC)* (pp. 7177–7182).

Showkatbakhsh, Mehrdad, Tabuada, Paulo, & Diggavi, Suhas (2016a). Secure system identification. In *54th annual allerton conference on communication, control, and computing* (pp. 1137–1141).

Showkatbakhsh, Mehrdad, Tabuada, Paulo, & Diggavi, Suhas (2016b). System identification in the presence of adversarial outputs. In *IEEE conference on decision and control (CDC)* (pp. 7177–7182).

Smith, Roy S. (2015). Covert misappropriation of networked control systems: Presenting a feedback structure. *Control Systems Magazine, IEEE*, *35*(1), 82–92.

Sundaram, Shreyas, Pajic, Miroslav, Hadjicostis, Christoforos N., Mangharam, Rahul, & Pappas, George J. (2010). The wireless control network: monitoring for malicious behavior. In *49th IEEE conference on decision and control (CDC)* (pp. 5979–5984).

Tiwari, Ashish, Dutertre, Bruno, Jovanović, Dejan, de Candia, Thomas, Lincoln, Patrick D., Rushby, John, et al. (2014). Safety envelope for security. In *ACM proceedings of the 3rd international conference on high confidence networked systems* (pp. 85–94).

Yong, Sze Zheng, Foo, Ming Qing, & Frazzoli, Emilio (2016). Robust and resilient estimation for cyber-physical systems under adversarial attacks. In *American control conference (ACC), 2016* (pp. 308–315).

Yoshikawa, T., & Bhattacharyya, S. (1975). Partial uniqueness: Observability and input identifiability. *IEEE Trans. Automat. Control*, *20*(5), 713–714.

Zhu, Minghui, & Martinez, Sonia (2014). On the performance analysis of resilient networked control systems under replay attacks. *IEEE Trans. Automat. Control*, *59*(3), 804–808.

Mehrdad Showkatbakhsh received his PhD in Electrical and Computer Engineering from the University of California, Los Angeles in 2019. His research lies in the area of cyber–physical security, privacy and its application to machine learning. He is a recipient of UCLA Henry Samueli and UCLA Preliminary Exam Fellowship. Mehrdad received his B.S. in Electrical Engineering from Sharif University, Tehran, Iran in 2013 and the M.S. degree in Electrical Engineering from University of California, Los Angeles, in 2015.

Yasser Shoukry is an Assistant Professor in the Department of Electrical Engineering and Computer Science at the University of California, Irvine. He received his Ph.D. in Electrical Engineering from the University of California, Los Angeles in 2015. He received the M.Sc. and the B.Sc. degrees (with distinction and honors) in Computer and Systems engineering from Ain Shams University, Cairo, Egypt in 2010 and 2007, respectively. Between September 2015 and July 2017, Yasser was a joint post-doctoral associate at UC Berkeley, UCLA, and UPenn. Before pursuing his Ph.D. at UCLA, he spent four years as an R&D engineer in the industry of automotive embedded systems. Yasser's research interests include the design and implementation of resilient Cyber–Physical Systems (CPS) and Internet-of-Things (IoT) by drawing on tools from embedded systems, formal methods, control theory, and machine learning.

Prof. Shoukry is the recipient of the NSF CAREER award, the Best Demo Award from the ACM/IEEE IPSN conference in 2017, the Best Paper Award from the ACM/IEEE ICCPS in 2016 and the Distinguished Dissertation Award from UCLA EE department in 2016. In 2015, he led the UCLA/Caltech/CMU team to win the NSF Early Career Investigators (NSFECI) research challenge. His team represented the NSF-ECI in the NIST Global Cities Technology Challenge, an initiative designed to advance the deployment of Internet of Things (IoT) technologies within a smart city. He is also the recipient of the 2019 George Corcoran Memorial Award for his contributions to teaching and educational leadership in the field of CPS and IoT.

Suhas N. Diggavi received the B. Tech. degree in electrical engineering from the Indian Institute of Technology, Delhi, India, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1998. After completing his Ph.D., he was a Principal Member Technical Staff in the Information Sciences Center, AT&T Shannon Laboratories, Florham Park, NJ. After that he was on the faculty of the School of Computer and Communication Sciences, EPFL, where he directed the Laboratory for Information and Communication Systems (LICOS). He is currently a Professor, in the Department of Electrical Engineering, at the University of California, Los Angeles, where he directs the Information Theory and Systems laboratory.

His research interests include information theory and its applications to several areas including wireless networks, cyber–physical systems, distributed computation and learning, security and privacy, genomics, data compression; more information can be found at http://licos.ee.ucla.edu. He has received several recognitions for his research including the 2013 IEEE Information Theory Society & Communications Society Joint Paper Award, the 2013 ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc) best paper award, the 2006 IEEE Donald Fink prize paper award. He served as a Distinguished Lecturer and also currently serves on board of governors for the IEEE Information theory society. He is a Fellow of the IEEE.

He has been an associate editor for IEEE Transactions on Information Theory, ACM/IEEE Transactions on Networking, IEEE Communication Letters, a guest editor for IEEE Selected Topics in Signal Processing and in the program committees of several IEEE conferences. He has also helped organize IEEE and ACM conferences including serving as the Technical Program Co-Chair for 2012 IEEE Information Theory Workshop (ITW), the Technical Program Co-Chair for the 2015 IEEE International Symposium on Information Theory (ISIT) and General co-chair for Mobihoc 2018. He has 8 issued patents.

Paulo Tabuada was born in Lisbon, Portugal, one year after the Carnation Revolution. He received his "Licenciatura" degree in Aerospace Engineering from Instituto Superior Tecnico, Lisbon, Portugal in 1998 and his Ph.D. degree in Electrical and Computer Engineering in 2002 from the Institute for Systems and Robotics, a private research institute associated with Instituto Superior Tecnico. Between January 2002 and July 2003 he was a postdoctoral researcher at the University of Pennsylvania. After spending three years at the University of Notre Dame, as an Assistant Professor, he joined the Electrical and Computer Engineering Department at the University of California, Los Angeles, where he currently is the Vijay K. Dhir Professor of Engineering.

Paulo Tabuada's contributions to cyber–physical systems have been recognized by multiple awards including the NSF CAREER award in 2005, the Donald P. Eckman award in 2009, the George S. Axelby award in 2011, the Antonio Ruberti Prize in 2015, the grade of fellow awarded by IEEE in 2017 and by IFAC in 2019. He has been program chair and general chair for several conferences in the areas of control and of cyber–physical systems such as NecSys, HSCC, and ICCPS. He currently serves as the chair of HSCC's steering committee and served on the editorial board of the IEEE Embedded Systems Letters and the IEEE Transactions on Automatic Control.