A Satisfiability Modulo Theory Approach to Secure State Reconstruction in Differentially Flat Systems Under Sensor Attacks

Yasser Shoukry, Pierluigi Nuzzo, Nicola Bezzo,
Alberto L. Sangiovanni-Vincentelli, Sanjit A. Seshia, and Paulo Tabuada

Abstract

We address the problem of estimating the state of a differentially flat system from measurements that may be corrupted by an adversarial attack. In cyber-physical systems, malicious attacks can directly compromise the system's sensors or manipulate the communication between sensors and controllers. We consider attacks that only corrupt a subset of sensor measurements. We show that the possibility of reconstructing the state under such attacks is characterized by a suitable generalization of the notion of s-sparse observability, previously introduced by some of the authors in the linear case. We also extend our previous work on the use of Satisfiability Modulo Theory solvers to estimate the state under sensor attacks to the context of differentially flat systems. The effectiveness of our approach is illustrated on the problem of controlling a quadrotor under sensor attacks.

I. Introduction

The increasing dependence on sensors and cyber components (e.g. digital processors and networks) to monitor and control a broad range of today's critical infrastructures is at the outset of unprecedented vulnerabilities and malicious attacks. A striking example of such attacks is the Stuxnet virus targeting SCADA systems [1]. In this attack, sensor measurements are replaced by previously recorded data, which, once they are fed to the controller, can lead to catastrophic situations. Other examples include the injection of false data in "smart" systems [2], and the non-invasive sensor spoofing attacks in the automotive domain [3].

To secure these cyber-physical systems, a possible strategy is to exploit an accurate mathematical model of the dynamics of the physical system under control, and analyze any discrepancy between the actual sensor measurements and the ones predicted by the model, to decide about the existence of an adversarial attack [4], [5]. Once the malicious sensors, if any, are detected and identified, it is then possible to estimate the actual system state by using the data collected from the attack-free sensors. In the following, we refer to this approach as *secure state reconstruction*.

The problem of state reconstruction in the presence of disturbances, in its general form, has attracted considerable attention from the control community over the years. Previous work addresses the problem in terms of robust filter (estimator) design against outliers [6], [7], [8]. However, the lack of a priori knowledge about the attack signals tends to limit the applicability of robust estimation techniques to security problems. In secure state reconstruction, no assumptions are usually made about the attacks, e.g., in terms of their stochastic properties, time evolution, or energy bounds.

A game theoretic formulation for the secure state reconstruction problem has been proposed in the literature, when the physical system is scalar [9]. An alternative reconstruction technique, still in the context of a scalar system equipped with one sensor, has also been derived based on the analysis of the performance degradation of a Kalman filter when the sensor is under attack [10]. Finally, the general case of a multidimensional system equipped with multiple sensors has been tackled [11], [5], [12], [13], [14], [15], [16], [17], [18] when the attackers are restricted to corrupt an unknown subset of the system sensors. However, all of the above contributions focus on problems for which the underlying dynamics can be described by a linear system.

Unlike previous work, we focus in this paper on the problem of secure state reconstruction for a class of nonlinear systems. Specifically, we consider physical systems whose dynamics can be described by a differentially flat system [19]. Differentially flat systems represent an important class of nonlinear systems, in that they encompass a wide range of mechanical systems, including several examples of ground and aerial vehicles.

While differentially flat systems can be converted into linear systems using dynamic feedback linearization and a change of coordinates, this technique would, however, require the knowledge of the system state. Since this is clearly our ultimate goal, it is not possible to directly apply the results from linear secure state reconstruction to differentially flat systems. We follow instead a different approach, by extending the notion of s-sparse observability [12], [13] from linear systems to nonlinear systems. Similarly to linear systems, we show that s-sparse observability provides a necessary and sufficient condition for the full reconstruction of the state regardless of the attack. Resting on this concept, we can then build on our previous work on

Y. Shoukry and P. Tabuada are with the Electrical Engineering Department, U.C. Los Angeles, {yshoukry, tabuada}@ucla.edu

P. Nuzzo, A. L. Sangiovanni-Vincentelli, and S. A. Seshia are with the Department of Electrical Engineering and Computer Sciences, U.C. Berkeley, {nuzzo,alberto,sseshia}@eecs.berkeley.edu

N. Bezzo is with the PRECISE Center, University of Pennsylvania, nicbezzo@seas.upenn.edu

This work was partially sponsored by the NSF award 1136174, by DARPA under agreement number FA8750-12-2-0247, by TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA, and by the NSF project ExCAPE: Expeditions in Computer Augmented Program Engineering (award 1138996). The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NSF, DARPA or the U.S. Government.

sound and complete secure state reconstruction for linear systems [15], [16], to develop an algorithm that can efficiently identify the corrupted sensors by leveraging Satisfiability Modulo Theory (SMT) solving [20] to tackle the combinatorial aspects of the problem. We illustrate our algorithm on the problem of controlling and stabilizing a quadrotor, while some of its sensors are under attack.

The rest of this paper is organized as follows. Section II formulates the secure state reconstruction problem. In Section III, we generalize the notion of s-sparse observability to nonlinear systems and then show that it is the necessary and sufficient condition to reconstruct the state in spite of the attack. Section IV presents the generalization of our previous SMT-based attack detection and reconstruction algorithm to differentially flat systems. In Section V, we discuss its application to the quadrotor case study. We finally draw some conclusions in Section VI.

II. The Secure State Reconstruction Problem

A. Notation

The symbols $\mathbb{N}, \mathbb{R}$ and $\mathbb{B}$ denote the sets of natural, real, and Boolean numbers, respectively. If S is a set, we denote by $|S|$ its cardinality. The support of a vector $x \in \mathbb{R}^n$, denoted by $\text{supp}(x)$, is the set of indices of the non-zero components of x. Similarly, the complement of the support of a vector x is denoted by $\overline{\text{supp}(x)} = \{1, \dots, n\} \setminus \text{supp}(x)$. We call a vector $x \in \mathbb{R}^n$ s-sparse, if x has s nonzero elements, i.e., if $|\text{supp}(x)| = s$.

Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a function given by $f(x) = (f_1(x), \dots, f_m(x))$, where $f_i : \mathbb{R}^n \to \mathbb{R}$ is the ith component of f. Then, for the set $\Gamma \subseteq \{1, \dots, m\}$, we denote by f_Γ the vector function obtained from f by removing all the components except those indexed by Γ. Similarly, $f_{\overline{\Gamma}}$ is obtained from f by removing the components indexed by Γ. Finally, we use the notation $\nabla_x f$ to denote the Jacobian matrix of f evaluated at x.

B. Dynamics and Attack Model

We consider a system of the form:

$$\Sigma_a \quad \begin{cases} x^{(t+1)} &= f\left(x^{(t)}, u^{(t)}\right), \\ y^{(t)} &= h\left(x^{(t)}\right) + a^{(t)} \end{cases} \tag{II.1}$$

where $x^{(t)} \in \mathcal{X} \subseteq \mathbb{R}^n$ is the system state, $u^{(t)} \in \mathcal{U} \subseteq \mathbb{R}^m$ is the system input, and $y^{(t)} \in \mathbb{R}^p$ is the observed output, all at time $t \in \mathbb{N}$. The map $f : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ represents the system dynamics. We will use the notation $f_u(x) = f(x, u)$ in the remainder of this paper. We also use the notation $f_{u_k u_{k-1} \dots u_1}(x)$ to denote the k-fold composition of f, i.e.,

$$f_{u_k u_{k-1} \dots u_1}(x) = f\left(f\left(f\left(u^{(1)}, x\right), \dots\right), u^{(k-1)}\right), u^{(k)}\right).$$

An attacker corrupts the sensor measurements y by either spoofing the sensor outputs, or by manipulating the data transmitted from the sensors to the controller. Independently of how the attack is implemented, its effect can be described by the s-sparse vector $a^{(t)} \in \mathbb{R}^p$. If sensor $i \in \{1, \dots, p\}$ is attacked then the ith component of $a^{(t)}$ is non-zero; otherwise the ith sensor is not attacked. Hence, s describes the number of attacked sensors. We make no assumptions on the vector $a^{(t)}$ other than being s-sparse. In particular, we do not assume bounds, statistical properties, or restrictions on the time evolution of the elements in $a^{(t)}$. While the value of s is not known, we assume the knowledge of an upper bound $\overline{s}$ on it.

C. Problem Formulation

Solving the secure state reconstruction problem implies estimating the state x from a set of measurements collected over a window of length $\tau \in \mathbb{N}$. Hence, we start by grouping the measurements from the ith sensor as:

$$Y_i^{(t)} = H_{u,i}\left(x^{(t-\tau+1)}\right) + E_i^{(t)} \tag{II.2}$$

where:

$$Y_i^{(t)} = \begin{bmatrix} y_i^{(t-\tau+1)} \\ y_i^{(t-\tau)} \\ \vdots \\ y_i^{(t)} \end{bmatrix}, E_i^{(t)} = \begin{bmatrix} a_i^{(t-\tau+1)} \\ a_i^{(t-\tau)} \\ \vdots \\ a_i^{(t)} \end{bmatrix}, H_{u,i}\left(x^{(t-\tau+1)}\right) = \begin{bmatrix} h_i\left(x^{(t-\tau+1)}\right) \\ h_i\left(f_{u_{t-\tau+1}}\left(x^{(t-\tau+1)}\right)\right) \\ \vdots \\ h_i\left(f_{u_t \dots u_{t-\tau+1}}\left(x^{(t-\tau+1)}\right)\right) \end{bmatrix}.$$

We then define:

$$Y^{(t)} = \begin{bmatrix} Y_1^{(t)} \\ \vdots \\ Y_p^{(t)} \end{bmatrix}, \quad E^{(t)} = \begin{bmatrix} E_1^{(t)} \\ \vdots \\ E_p^{(t)} \end{bmatrix}, \quad H_u = \begin{bmatrix} H_{u,1} \\ \vdots \\ H_{u,p} \end{bmatrix}$$

where, with some abuse of notation, Y_i, E_i and $H_{u,i}$ are used to denote the ith block of $Y^{(t)}, E^{(t)}$ and H_u, respectively. Using the same notation, we denote by Y_Γ, E_Γ, and $H_{u,\Gamma}$ the blocks indexed by the elements in the set Γ. Moreover, for simplicity, we drop the time t argument in the following, since we assume that the secure state reconstruction problem is to be solved at every time instance.

Let (x^*, E^*) denote the actual state of the system and the actual attack vector. Let also $b^* \in \mathbb{B}^p$ be a vector of binary indicator variables such that $b_i^* = 0$ when the ith sensor is attack-free and $b_i = 1$ otherwise. It follows from (II.2) that:

$$Y_i = \begin{cases} H_{u,i}(x^*) & \text{if } b_i^* = 0 \\ H_{u,i}(x^*) + E_i^* & \text{if } b_i^* = 1. \end{cases}$$

Therefore, we are interested in a state estimate x and a vector of binary indicator variables $b = (b_1, \ldots, b_p)$ such that the discrepancy between the collected measurements Y_i and the expected outputs $H_{u,i}(x)$ is zero for all the sensors that are labeled as attack-free sensors ($b_i^* = 0$). Furthermore, the estimated state x should be equal to x^*. These requests can be formalized as follows.

Problem II.1. ***(Secure State Reconstruction)*** *For the control system under attack Σ_a (defined in* (II.1)*), construct the estimate $\eta = (x, b) \in \mathbb{R}^n \times \mathbb{B}^p$ such that $\eta \models \phi$ (i.e., η satisfies the formula ϕ), where:*

$$\phi ::= \bigwedge_{i=1}^{p} \left(\neg b_i \Rightarrow \|Y_i - H_{u,i}(x)\|_2^2 = 0 \right) \bigwedge \left(\sum_i^p b_i \leq \overline{s} \right), \tag{II.3}$$

subject to $(x^* = x) \wedge (\text{supp}(b^*) \subseteq \text{supp}(b))$.

The second clause in the formula ϕ rules out the trivial solution in which all sensors are labelled as attacked, by enforcing a cardinality constraint on the number of attacked sensors, which is required to be bounded by $\overline{s}$.

As in the case of linear systems [16], the secure state reconstruction problem formulation in Problem II.1 does not ask for a solution with the minimal number of attacked sensors. However, as shown in [16], it is possible to obtain the minimal set of sensors under attack by invoking a solver for Problem II.1 multiple times. In the next section, we characterize the conditions that guarantee the existence of a solution for this problem.

III. s-Sparse Observability

For linear systems, the notion of s-sparse observability plays a key role in determining the existence of a solution for Problem II.1 [12]. In this section, we generalize this notion to the case of nonlinear systems. To do so, we consider an attack-free discrete-time nonlinear system of the form:

$$\Sigma \quad \begin{cases} x^{(t+1)} & = f\left(x^{(t)}, u^{(t)}\right), t \in \mathbb{N} \\ y^{(t)} & = h\left(x^{(t)}\right) \end{cases} \tag{III.1}$$

and recall the general definitions of indistinguishability and observability.

Definition III.1 (Indistinguishability)**.** *We say that two states $x, x' \in \mathcal{X}$ of system Σ are* indistinguishable *from measurements collected from the set of sensors indexed by Γ over a window of length τ, and we write $x \; I_\Gamma^\tau \; x'$ if, for every sequence of controls $u^{(1)}, \ldots, u^{(\tau)} \in \mathbb{R}^m$ we have $H_{u,\Gamma}(x) = H_{u,\Gamma}(x')$.*

Definition III.2 (Observability)**.** *A state $x \in \mathcal{X}$ of system Σ is said to be* observable *using measurements collected from the set of sensors indexed by Γ over a window of length τ $\left((\tau, \Gamma)\text{-observable for short}\right)$, if, for each $x' \in \mathcal{X}$, $x \; I_\Gamma^\tau \; x'$ implies $x = x'$.*

Definition III.3. *A system Σ is said to be* (τ, Γ)-observable *if all the states $x \in \mathcal{X}$ are (τ, Γ)-observable.*

We can now define the notion of s-sparse observability as follows.

Definition III.4 (**s-Sparse τ-Observable System**)**.** *The nonlinear control system Σ, defined in* (III.1)*, is said to be s-sparse τ-observable if for every set $\Gamma \subseteq \{1, \ldots, p\}$ with $|\Gamma| = s$, the system Σ is $(\tau, \overline{\Gamma})$-observable, where $\overline{\Gamma} = \{1, \ldots, p\} \setminus \Gamma$.*

In other words, a system is s-sparse τ-observable if it is τ-observable from any choice of $(p - s)$ sensors.

A. s-Sparse Observability and Secure State Reconstruction

Similarly to linear systems, we show that $2\overline{s}$-sparse observability is a necessary and sufficient condition for the existence of a solution of the secure state reconstruction problem. To do so, we start by showing an intermediate result, stating that the correct reconstruction of the system state implies correct estimation of the attack support. To simplify the notation, we also drop the subscript u in all of the proofs.

Proposition III.5. *For any pair $\eta = (x, b)$ that satisfies ϕ in* (II.3) *(i.e., such that $\eta \models \phi$), the following holds:*

$$x^* = x \Rightarrow \textit{supp}(b^*) \subseteq \textit{supp}(b).$$

Proof: The result follows from the fact that the first set of clauses in ϕ enforces:

$$\begin{aligned}\bigwedge_{i \in \overline{\text{supp}(b)}} \|Y_i - H_i(x)\|_2^2 = 0 \Rightarrow \left\|Y_{\overline{\text{supp}(b)}} - H_{\overline{\text{supp}(b)}}(x)\right\|_2^2 &= 0 \\ \Rightarrow \left\|H_{\overline{\text{supp}(b)}}(x^*) + E^*_{\overline{\text{supp}(b)}} - H_{\overline{\text{supp}(b)}}(x)\right\|_2^2 &= 0 \\ \Rightarrow \left\|E^*_{\overline{\text{supp}(b)}}\right\|_2^2 &= 0\end{aligned}$$

where the last implication follows from x and x^* being equal. The last equality implies that the attack vector E^*, after removing the sensors indexed by the estimate set $\text{supp}(b)$, is equal to zero; hence, we conclude that $\text{supp}(b^*) \subseteq \text{supp}(b)$. ■

Now we can state the main result of this section as follows.

Theorem III.6. *For any pair $\eta = (x, b)$ that satisfy ϕ in* (II.3) *the following holds:*

$$x^* = x \qquad \wedge \qquad \textit{supp}(b^*) \subseteq \textit{supp}(b)$$

if and only if the dynamical system Σ_a defined by (II.1) *is $2\overline{s}$-sparse τ-observable.*

Proof: We need to show that $2\overline{s}$-sparse τ-observability is a necessary and sufficient condition for $x = x^*$ to hold. Once this is established, $\text{supp}(b^*) \subseteq \text{supp}(b)$ follows from Proposition III.5.

We assume for the sake of contradiction that there exists a pair $\eta = (x, b)$ that satisfies ϕ, and such that $x \neq x^*$, while the system Σ_a is $2\overline{s}$-sparse τ-observable. Then, the first set of clauses in ϕ implies that:

$$\begin{aligned}\bigwedge_{i \in \overline{\text{supp}(b)}} \|Y_i - H_i(x)\|_2^2 = 0 \Rightarrow \left\|Y_{\overline{\text{supp}(b)}} - H_{\overline{\text{supp}(b)}}(x)\right\|_2^2 &= 0 \\ \Rightarrow \left\|Y_{\overline{\text{supp}(b)\cup\text{supp}(b^*)}} - H_{\overline{\text{supp}(b)\cup\text{supp}(b^*)}}(x)\right\|_2^2 &= 0 \\ \overset{(a)}{\Rightarrow} \left\|H_{\overline{\text{supp}(b)\cup\text{supp}(b^*)}}(x^*) - H_{\overline{\text{supp}(b)\cup\text{supp}(b^*)}}(x)\right\|_2^2 &= 0 \\ \Rightarrow H_{\overline{\text{supp}(b)\cup\text{supp}(b^*)}}(x^*) = H_{\overline{\text{supp}(b)\cup\text{supp}(b^*)}}(x) &\qquad \text{(III.2)}\end{aligned}$$

where the implication (a) follows by using the equality (II.2), along with the fact that $E^*_{\overline{\text{supp}(b^*)}}$ being zero implies that $E^*_{\overline{\text{supp}(b)\cup\text{supp}(b^*)}}$ is also equal to zero.

However, by the second clause in ϕ we know that the cardinality of $\text{supp}(b)$ is at most $\overline{s}$, hence the cardinality of $|\text{supp}(b) \cup \text{supp}(b^*)|$ is at most $2\overline{s}$. Equality (III.2) would then imply that x and x^* are indistinguishable using $p - 2\overline{s}$ sensors, which in turn implies that the system Σ_a is not $2\overline{s}$-sparse τ-observable, a contradiction with respect to our original assumption.

Conversely, if the system Σ_a is not $2s$-sparse τ-observable then there exists a vector x with $x \neq x^*$ which is indistinguishable from x^*, when using $p - 2\overline{s}$ sensors. This in turn implies that $H_\Gamma(x^*) = H_\Gamma(x)$ for some Γ that have cardinality $p - 2\overline{s}$. Hence, we can define two indicator variables b and b^* such that $\Gamma = \overline{\text{supp}(b) \cup \text{supp}(b^*)}$, $|\text{supp}(b)| = \overline{s}$ and $|\text{supp}(b^*)| = \overline{s}$. We can also define the attack vector E^* as :

$$E^*_{\text{supp}(b^*)} = H_{\text{supp}(b^*)}(x) - H_{\text{supp}(b^*)}(x^*).$$

Then, using the above definitions, together with the assumption $H_\Gamma(x^*) = H_\Gamma(x)$, we obtain:

$$\begin{aligned}\left\|Y_{\overline{\text{supp}(b)}} - H_{\overline{\text{supp}(b)}}(x)\right\|_2^2 &= \left\|H_{\overline{\text{supp}(b)}}(x^*) + E^*_{\overline{\text{supp}(b)}} - H_{\overline{\text{supp}(b)}}(x)\right\|_2^2 \\ &= \left\|\begin{bmatrix} H_\Gamma(x^*) \\ H_{\text{supp}(b^*)}(x^*)\end{bmatrix} + \begin{bmatrix} 0 \\ E^*_{\text{supp}(b^*)}\end{bmatrix} - \begin{bmatrix} H_\Gamma(x) \\ H_{\text{supp}(b^*)}(x)\end{bmatrix}\right\|_2^2 \\ &= 0\end{aligned}$$

which implies that there exists an estimate $\eta = (x, b)$, with $x \neq x^*$ that also satisfies ϕ. ■

B. Differential Flatness and s-Sparse Flatness

For the rest of this paper, we consider a special class of nonlinear systems known as *differentially flat*[1] systems. A system is differentially flat if the state and the input can be reconstructed from current and previous outputs. More formally,

Definition III.7 (Differential Flatness)**.** *System is differentially flat if there exist an integer $k \in \mathbb{N}$, and functions α and β, such that the state and the input can be reconstructed from the outputs $y^{(t)}$ as follows:*

$$x^{(t)} = \alpha\left(y^{(t)}, y^{(t-1)}, \ldots, y^{(t-k+1)}\right) \tag{III.3}$$

$$u^{(t)} = \beta\left(y^{(t)}, y^{(t-1)}, \ldots, y^{(t-k+1)}\right). \tag{III.4}$$

In such case, the output $y^{(t)}$ is called a flat output.

In the remainder of this paper, we assume that the window length τ in (II.2) is chosen such that $\tau = k$.

Definition III.8 (**s-Sparse Flat System**)**.** *The nonlinear control system Σ_a, defined by* (II.1)*, is said to be s-sparse flat if for every set $\Gamma \subseteq \{1, \ldots, p\}$ with $|\Gamma| = s$, the system $\Sigma_{\overline{\Gamma}}$:*

$$\Sigma_{\overline{\Gamma}} \quad \begin{cases} x^{(t+1)} & = f\left(x^{(t)}, u^{(t)}\right), \\ y^{(t)} & = h_{\overline{\Gamma}}\left(x^{(t)}\right) \end{cases} \tag{III.5}$$

is differentially flat.

In other words, the system is s-sparse flat if any choice of $p - s$ sensors is a flat output. It is then straightforward to show that s-sparse flatness implies s-sparse τ-observability.

IV. Secure State Reconstruction Using SMT Solving

The secure state reconstruction problem is combinatorial, since a direct solution would require constructing the state from all different combinations of $p - \overline{s}$ sensors to determine which sensors are under attack. In this section, we show how using SMT solving can dramatically reduce the complexity of the reconstruction algorithm.

To decide whether the combination of Boolean and nonlinear constraints in (II.3) is satisfiable, we develop the detection algorithm Imhotep-SMT using the *lazy* SMT paradigm [20]. By building upon the Imhotep-SMT solver [16], [21], our decision procedure combines a SAT solver (SAT-Solve) and a theory solver ($\mathcal{T}$-Solve). However, differently than [16], [21], the theory solver in this paper can also reason about the nonlinear constraints in (II.3), as generated from a differentially flat system. The SAT solver efficiently reasons about combinations of Boolean and pseudo-Boolean constraints[2], using the David-Putnam-Logemann-Loveland (DPLL) algorithm [22] to suggest possible assignments for the nonlinear constraints. The theory solver checks the consistency of the given assignments, and provides the reason for the conflict, a *certificate*, or a counterexample, whenever inconsistencies are found. Each certificate results in learning new constraints which will be used by the SAT solver to prune the search space. The complex decision task is thus broken into two simpler tasks, respectively, over the Boolean and nonlinear domains.

A. Overall Architecture

As illustrated in Algorithm 1, we start by mapping each nonlinear constraint to an auxiliary Boolean variable c_i to obtain the following (pseudo-)Boolean satisfiability problem:

$$\phi_B := \bigwedge_{i=1}^{p} \left(\neg b_i \Rightarrow c_i\right) \bigwedge \left(\sum_{i=1}^{p} b_i \leq \overline{s}\right)$$

where $c_i = 1$ if $\|Y_i - H_{u,i}(x)\|_2 = 0$ is satisfied, and zero otherwise. By only relying on the Boolean structure of the problem, SAT-Solve returns an assignment for the variables b_i and c_i (for $i = 1, \ldots, p$), thus hypothesizing which sensors are attack-free, hence which nonlinear constraints should be jointly satisfied. This Boolean assignment is then used by $\mathcal{T}$-Solve to determine whether there exists a state $x \in \mathbb{R}^n$ which satisfies all the nonlinear constraints related to the unattacked sensors, i.e. $\{\|Y_i - H_{u,i}(x)\|_2 = 0 | i \in \overline{\text{supp}(b)}\}$ is the set of constraints sent to $\mathcal{T}$-Solve. If x is found, Imhotep-SMT terminates with `SAT` and provides the solution (x, b). Otherwise, the `UNSAT` certificate ϕ_{cert} is generated in terms of new Boolean constraints, explaining which sensor measurements are conflicting and may be under attack. A naïve certificate can always be generated in the form of:

$$\phi_{\text{triv-cert}} = \sum_{i \in \overline{\text{supp}(b)}} b_i \geq 1, \tag{IV.1}$$

[1]Although the term *difference flatness* is sometimes used in the literature for systems governed by difference equations, we choose to employ the widely accepted term *differential flatness*.

[2]A pseudo-Boolean constraint is a linear constraint over Boolean variables with integer coefficients.

Algorithm 1 IMHOTEP-SMT

status := UNSAT;
$\phi_B := \bigwedge_{i=1}^{p} (\neg b_i \Rightarrow c_i) \bigwedge (\sum_{i=1}^{p} b_i \leq \overline{s})$;
while status == UNSAT **do**
  $(b, c) :=$ SAT-SOLVE(ϕ_B);
  (status, x) := $\mathcal{T}$-SOLVE.CHECK($\overline{\text{supp}}(b)$);
  if status == UNSAT **then**
    $\phi_{\text{cert}} := \mathcal{T}$-SOLVE.CERTIFICATE($b, x$);
    $\phi_B := \phi_B \wedge \phi_{\text{cert}}$;
  end if
end while
return $\eta = (x, b)$;

which encodes the fact that at least one of the sensors in the set $\overline{\text{supp}(b)}$ (i.e. for which $b_i = 0$ in the current iteration) is actually under attack, and must be set to one in the next assignment of the SAT solver. The augmented Boolean problem is then fed back to SAT-SOLVE to produce a new assignment, and the sequence of new SAT queries repeats until $\mathcal{T}$-SOLVE terminates with SAT.

By assuming that the system is $2\overline{s}$-sparse flat, it follows from Theorem III.6 that there always exists a solution to Problem II.1, hence Algorithm 1 will always terminate. However, to help the SAT solver quickly converge towards the correct assignment, a central problem in lazy SMT solving is to generate succinct explanations whenever conjunctions of nonlinear constraints are unfeasible.

The rest of this section will then focus on the implementation of the two main tasks of $\mathcal{T}$-SOLVE, namely, (i) checking the satisfiability of a given assignment ($\mathcal{T}$-SOLVE.CHECK), and (ii) generating succinct UNSAT certificates ($\mathcal{T}$-SOLVE.CERTIFICATE).

B. Satisfiability Checking

It follows from the $2\overline{s}$-sparse flatness property discussed in Section II, that for a given assignment of the Boolean variables b, with $|\text{supp}(b)| \leq \overline{s}$, the remaining $p - \overline{s}$ sensors define a flat output as:

$$y_{\mathcal{I}}^{(t)}, \quad \ldots, \quad y_{\mathcal{I}}^{(t-\tau+1)}$$

where $\mathcal{I} = \overline{\text{supp}(b)}$. The next step is to use the flat output in order to calculate the estimate $x = \alpha\left(y_{\mathcal{I}}^{(t)}, \ldots, y_{\mathcal{I}}^{(t-\tau+1)}\right)$. Finally, we evaluate if the condition:

$$\left\| Y_{\overline{\text{supp}(b)}} - H_{u,\overline{\text{supp}(b)}}(x) \right\|_2^2 = 0 \tag{IV.2}$$

is satisfied. This procedure is summarized in Algorithm 2.

Algorithm 2 $\mathcal{T}$-SOLVE.CHECK($\mathcal{I}$)

Construct the state estimate:
  $x := \alpha\left(y_{\mathcal{I}}^{(t)}, \ldots, y_{\mathcal{I}}^{(t-\tau+1)}\right)$
if $\left\| Y_{\mathcal{I}} - H_{u,\mathcal{I}(x)} \right\|_2 == 0$ **then**
  status := SAT;
else
  status := UNSAT;
end if
return (status, x)

C. Generating Succinct UNSAT Certificates

Whenever $\mathcal{T}$-SOLVE.CHECK provides UNSAT, the naïve certificate can always be generated as in (IV.1). However, such trivial certificate does not provide much information, since it only excludes the current assignment from the search space, and can lead to exponential execution time, as reflected by the following proposition.

Proposition IV.1. *Let the linear dynamical system Σ_a defined in* (II.1) *be $2\overline{s}$-sparse observable. Then, Algorithm 1 which uses the trivial UNSAT certificate $\phi_{\text{triv-cert}}$ in* (IV.1) *returns $\eta = (x, b)$ such that:*

$$x^* - x \quad \wedge \quad \textit{supp}(b^*) \subseteq \textit{supp}(b),$$

where x^* and b^* are the actual system state and attack indicator vector, as defined in Section II-C. Moreover, the upper bound on the number of iterations of Algorithm 1 is $\sum_{s=0}^{\overline{s}} \binom{p}{s}$.

Proof: Correctness of Algorithm 1 follows directly from the $2\overline{s}$-sparse flatness along with Theorem III.6. The worst case bound on the number of iterations would happen when the solver exhaustively explores all possible combinations of attacked sensors with cardinality less than or equal to $\overline{s}$ in order to find the correct assignment. This is equal to $\sum_{s=0}^{\overline{s}} \binom{p}{s}$ iterations. ■

The generated UNSAT certificate heavily affects the overall execution time of Algorithm 1: the smaller the certificate, the more information is learnt and the faster is the convergence of the SAT solver to the correct assignment. For example, a certificate with $b_i = 1$ would identify exactly one attacked sensor at each step. Therefore, our objective is to design an algorithm that can lead to more *compact certificates* to enhance the execution time of IMHOTEP-SMT. To do so, we exploit the specific structure of the secure state reconstruction problem and generate customized, yet stronger, UNSAT certificates.

First, we observe that the measurements of each sensor $Y_i = H_{u,i}(x)$ define a set $\mathbb{H}_i \subseteq \mathcal{X}$ as:

$$\mathbb{H}_i = \{x \in \mathcal{X} \mid Y_i - H_{u,i}(x) = 0\}.$$

It is then straightforward to show the following result.

Proposition IV.2. *Let the nonlinear dynamical system Σ_a defined in* (II.1) *be $2\overline{s}$-sparse flat. Then, for any set of indices $\mathcal{I} \subseteq \{1, \dots, p\}$, the following statements are equivalent:*

- *$\mathcal{T}$-SOLVE.CHECK($\mathcal{I}$) returns* `UNSAT`*,*
- *$\bigcap_{i\in\mathcal{I}} \mathbb{H}_i = \emptyset$.*

The existence of a compact Boolean constraint that explains a conflict is then guaranteed by the following Lemma.

Lemma IV.3. *Let the nonlinear dynamical system Σ_a defined in* (II.1) *be $2\overline{s}$-sparse flat. If $\mathcal{T}$-SOLVE.CHECK($\mathcal{I}$) is* `UNSAT` *for a set $\mathcal{I}$, with $|\mathcal{I}| > p - 2\overline{s}$, then there exists a subset $\mathcal{I}_{temp} \subset \mathcal{I}$ with $|\mathcal{I}_{temp}| \leq p - 2\overline{s} + 1$ such that $\mathcal{T}$-SOLVE.CHECK($\mathcal{I}_{temp}$) is also* `UNSAT`*.*

Proof: Consider any set of sensors $\mathcal{I}' \subset \mathcal{I}$ such that $|\mathcal{I}'| = p - 2\overline{s}$ and $\bigcap_{i\in\mathcal{I}'} \mathbb{H}_i$ is not empty. If such set $\mathcal{I}'$ does not exist, then the result follows trivially. If the set $\mathcal{I}'$ exists, then:

$$\bigcap_{i\in\mathcal{I}'} \mathbb{H}_i \neq \emptyset \Rightarrow \|Y_{\mathcal{I}'} - H_{u,\mathcal{I}'}(x')\|_2^2 = 0 \quad \forall x' \in \bigcap_{i\in\mathcal{I}'} \mathbb{H}_i$$

However, since the cardinality of $\mathcal{I}'$ is equal to $p - 2\overline{s}$, it follows from the $2\overline{s}$-sparse flatness that any state $x' \in \bigcap_{i\in\mathcal{I}'} \mathbb{H}_i$ is distinguishable using $p - 2\overline{s}$ sensors, for which we conclude that the intersection $\bigcap_{i\in\mathcal{I}'} \mathbb{H}_i$ is a single point, named x'. Now, since $\mathcal{T}$-SOLVE.CHECK($\mathcal{I}$) is `UNSAT`, it follows from Proposition IV.2 that:

$$\bigcap_{i\in\mathcal{I}} \mathbb{H}_i = \emptyset \Rightarrow \bigcap_{i\in\mathcal{I}'} \mathbb{H}_i \cap \bigcap_{i\in\mathcal{I}\setminus\mathcal{I}'} \mathbb{H}_i = \emptyset\{x'\} \cap \bigcap_{i\in\mathcal{I}\setminus\mathcal{I}'} \mathbb{H}_i = \emptyset,$$

which in turn implies that there exists at least one sensor $i \in \mathcal{I} \setminus \mathcal{I}'$ such that its set $\mathbb{H}_i$ does not include the point x'. Now, we define $\mathcal{I}_{temp}$ as $\mathcal{I}_{temp} = \mathcal{I}' \cup i$ and note that $|\mathcal{I}_{temp}| \leq p - 2\overline{s} + 1$, which concludes the proof. ■

Based on the intuition in the proof of Lemma IV.3, our algorithm works as follows. First, we construct the set of indices $\mathcal{I}'$ by picking any random set of $p - 2\overline{s}$ sensors. We then search for one additional sensor i' which can lead to a conflict with the sensors indexed by $\mathcal{I}'$. To do this, we call $\mathcal{T}$-SOLVE.CHECK by passing the set $\mathcal{I}_{temp} := \mathcal{I}' \cup i'$ as an argument. If the check returned `SAT`, then we label these sensors as "non-conflicting" and we repeat the same process by replacing the sensor indexed by i' with another sensor until we reach a conflicting set. It then follows from Lemma IV.3 that this process terminates revealing a collection of $p - 2\overline{s} + 1$ conflicting sets. Once this collection is discovered, we stop by generating the following, more compact, certificate:

$$\phi_{\text{cert}} := \sum_{i\in\mathcal{I}_{temp}} b_i \geq 1.$$

Although the prescribed process will always terminate regardless of the selection of the initial set $\mathcal{I}'$ or the order followed to select i', the execution time may change. In Algorithm 3, we implement a heuristic for the selection of the initial set $\mathcal{I}'$ and the succeeding indexes, inspired by the strategy we have adopted in the context of linear systems [16]. We are now ready to state the main result of this section.

Theorem IV.4. *Let the nonlinear dynamical system Σ_a defined in* (II.1) *be $2\overline{s}$-sparse flat. Then, Algorithm 1 using the conflicting UNSAT certificate ϕ_{cert} in Algorithm 3 returns $\eta = (x, b)$ such that:*

$$x^* - x \quad \wedge \quad \mathit{supp}(b^*) \subseteq \mathit{supp}(b)$$

Moreover, the upper bound on the number of iterations of Algorithm 1 is $\binom{p}{p-2\overline{s}+1}$.

Algorithm 3 $\mathcal{T}$-SOLVE.CERTIFICATE$(\mathcal{I}, x)$

Compute normalized residuals
  $r := \bigcup_{i\in\mathcal{I}} \{r_i\}$,
    $r_i := \|Y_i - H_{u,i}(x)\|_2^2 \,/\, \|\nabla_x H_{u,i}\|_2^2,\ i \in \mathcal{I}$;
Sort the residual variables
  $r\_sorted :=$ sortAscendingly(r);
Pick the index corresponding to the maximum residual
  $\mathcal{I}\_max\_r :=$ Index$(r_sorted_{\{|\mathcal{I}|,|\mathcal{I}|-1,\ldots,p-2\overline{s}+1\}})$;
  $\mathcal{I}\_min\_r :=$ Index$(r_sorted_{\{1,\ldots,p-2\overline{s}\}})$;
Search linearly for the UNSAT certificate
  status = SAT; counter $= 1$;
  $\mathcal{I}\_temp := \mathcal{I}\_min\_r \cup \mathcal{I}\_max\_r_{counter}$;
while status == SAT **do**
  (status, x) := $\mathcal{T}$-SOLVE.CHECK$(\mathcal{I}_temp)$;
  if status == UNSAT **then**
    $\phi_{\text{cert}} := \sum_{i\in\mathcal{I}\_temp} b_i \geq 1$;
  else
    counter := counter + 1;
    $\mathcal{I}\_temp := \mathcal{I}\_min\_r \cup \mathcal{I}\_max\_r_{counter}$;
  end if
end while
return ϕ_{cert}

Proof: Correctness follows from Theorem III.6 along with the $2\overline{s}$-flatness condition. The upper bound on the number of iterations of Algorithm 1 can be derived as follows. First, Lemma IV.3 ensures that each certificate ϕ_{cert} has at most $p-2\overline{s}+1$ variables. Since we know that the algorithm always terminates, the worst case would then happen when the solver exhaustively generates all the conflicting sets of cardinality $p-2\overline{s}+1$. This leads to a number of iterations equal to $\binom{p}{p-2\overline{s}+1}$. ■

V. CASE STUDY: SECURING A QUADROTOR MISSION

We demonstrate the effectiveness of our detection algorithm by applying it to a waypoint navigation mission for a quadrotor unmanned aerial vehicle (UAV), in which the UAV needs to cross a workspace from a starting point to a desired goal. The dynamical model of the quadrotor and its controller, based on [23], [24], are summarized below.

A. Dynamical Model

The dynamical model of the quadrotor consists of twelve states which are:

- $\mathbf{p} = (p_x, p_y, p_z)$ is the quadrotor center of mass in the inertial frame W (world frame).
- $\mathbf{v} = (v_x, v_y, v_z)$ is the quadrotor linear velocity.
- $\vartheta = (\vartheta_\phi, \vartheta_\theta, \vartheta_\psi)$ is the quadrotor orientation (or attitude) expressed in terms of Euler angles: roll, pitch, yaw.
- $\omega = (\omega_\phi, \omega_\theta, \omega_\psi)$ is the quadrotor angular velocity.

The quadrotor is equipped with four motors. The thrust produced by the ith motor is denoted by F_i and is directly proportional to the rotor speed. The resulting vertical force is denoted by u_1 and is equal to:

$$u_1 = F_1 + F_2 + F_3 + F_4 - mg$$

where m is the mass and g is the gravitational acceleration. The motors' thrust also induce three moments, on the quadrotor center of mass, denoted by u_2, u_3, u_4. These moments can be computed from the thrusts as follows:

$$\begin{pmatrix} u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} l_{1x} & -l_{2x} & l_{3x} & -l_{4x} \\ -l_{1y} & l_{2y} & l_{3y} & -l_{4y} \\ \mu & \mu & -\mu & -\mu \end{pmatrix} \begin{pmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{pmatrix}$$

where l_{ix} and l_{iy} are the x and y components of the distance between the motor i and the center of the quadrotor, respectively, and μ is a constant representing the relationship between lift and drag.

B. Controller

The controller is derived by linearizing the equations of motion and the motor models at an operating point that corresponds to the nominal hover state, i.e. $\mathbf{p} = (p_x, p_y, p_z)$, $\vartheta = (0, 0, \psi)$, $\mathbf{v} = (0, 0, 0)$ and $\omega = (0, 0, 0)$ under the assumptions of small roll and pitch angles, which leads to $\cos(\vartheta_\phi) = \cos(\vartheta_\theta) \approx 1$, $\sin(\vartheta_\phi) \approx \phi$, $\sin(\vartheta_\theta) \approx \vartheta_\theta$. The nominal values for the inputs at hover are $u_1 = mg$, $u_2 = u_3 = u_4 = 0$.

To control the quadrotor to follow a desired trajectory, we use a two-level decoupled control scheme consisting of a low-level attitude control, which usually runs at 1KHz, and a high-level position control running at a lower rate 50Hz. The position control is used to track a desired trajectory characterized by $\mathbf{p}^{\text{des}}$ and ϑ^{des}. Using a PID feedback controller we can then control the position and velocity of the quadrotor to maintain the desired trajectory. Similarly, we can realize the attitude control using PD controllers.

C. Securing the Quadrotor Trajectory

The quadrotor is equipped with a GPS measuring the position vector and two inertial measurement units (IMUs), whose outputs are fused to generate an estimate for the body angular and linear velocities. We numerically simulate the model of the quadrotor and the controller. In our scenario, the quadrotor goal is to takeoff vertically and then move along a square trajectory. However, one of the IMU's output, the vertical velocity sensor, is attacked by injecting a sinusoidal signal on top of the actual sensor readings. As shown in Fig. 1 (bottom), the attack is injected after the quadrotor has completed the takeoff maneuver, and only along two parallel sides of the whole square trajectory.

To implement the secure state reconstruction algorithm, IMHOTEP-SMT uses an approximate discretized model of the plant along with the sensor measurements. To discretize the model we use the same sampling time ($T_s = 20$ ms) of the controller. Moreover, we adopt a first-order forward Euler approximation scheme which preserves the differential flatness of the original system. To accommodate the model mismatch due to the discrete approximation, as well as round-off errors, we replace the condition in line 2 of Algorithm 2 with $\|Y_{\mathcal{I}} - H_{u,\mathcal{I}}(x)\|_2 \leq \epsilon$, where we set ϵ to 0.1 in our experiments.

Fig. 1(a) (top) shows the effect of the attack when the quadrotor operates without secure state reconstruction algorithm. As evident from the two corners of the square trajectory corresponding to coordinates $(2.5, 0, 1)$ and $(0, 2.5, 1)$, the injected attack harmfully impairs the stability of the quadrotor, due to incorrect state reconstruction, as shown in Fig. 1(a) (middle). Fig. 1(b) (top) shows instead the trajectory of the quadrotor when operated using IMHOTEP-SMT to perform secure state reconstruction. The estimation error on v_z produced by IMHOTEP-SMT is in the order of 10^{-2} m/s, and always bounded, where the bound depends on the error due to mismatch between the model used for estimation and the actual quadrotor dynamics (the controller is designed to be robust against bounded perturbations). The state and the support of the attack are correctly estimated also in the presence of model mismatch: the quadrotor is able to follow the required trajectory and achieve its goal. Finally, the average execution time of 16.1 ms (smaller than the 20ms sampling time of the position controller) on an Intel Core i7 3.4-GHz processor with 8 GB of memory, is compatible with several real-time applications.

VI. Conclusions

We have investigated, for the first time, the state reconstruction problem from a set of adversarially attacked sensors for a class of nonlinear systems, namely differentially flat systems. Given an upper bound $\overline{s}$ on the number of attacked sensors, we showed that $2\overline{s}$-observability is a necessary and sufficient condition for reconstructing the state in spite of the attack. We have then proposed a Satisfiability Modulo Theory based detection algorithm for differentially flat systems, by extending our previous results, reported in [15], [16], to differentially flat systems. Numerical results show that secure state estimation in complex nonlinear systems, such as in waypoint navigation of a quadrotor under sensor attacks, can indeed be performed with our algorithm in an accurate and efficient way.

References

[1] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Security and Privacy Magazine*, vol. 9, no. 3, pp. 49–51, 2011.

[2] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proceedings of the 16th ACM conference on Computer and communications security*, ser. CCS '09. New York, NY, USA: ACM, 2009, pp. 21–32.

[3] Y. Shoukry, P. D. Martin, P. Tabuada, and M. B. Srivastava, "Non-invasive spoofing attacks for anti-lock braking systems," in *Workshop on Cryptographic Hardware and Embedded Systems*, ser. G. Bertoni and J.-S. Coron (Eds.): CHES 2013, LNCS 8086. International Association for Cryptologic Research, 2013, pp. 55–72.

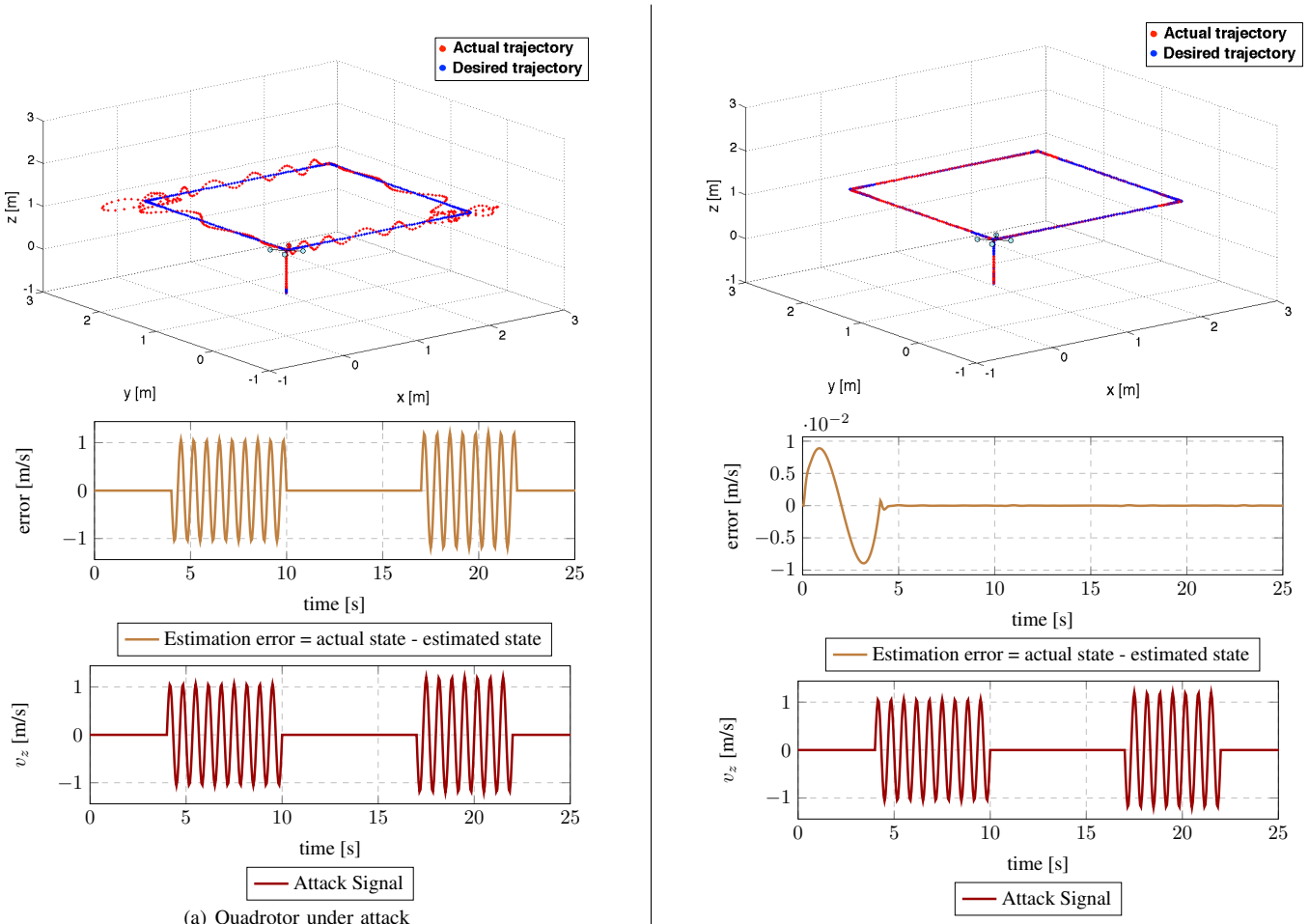
[4] F. Pasqualetti, F. Dorfler, and F. Bullo, "Control-theoretic methods for cyber-physical security," *IEEE Control Systems Magazine*, Aug. 2014, to appear. [Online]. Available: http://motion.me.ucsb.edu/pdf/2013u-pdb.pdf

[5] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1454–1467, June 2014.

[6] J. Mattingley and S. Boyd, "Real-time convex optimization in signal processing," *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 50–61, May 2010.

[7] I. C. Schick and S. K. Mitter, "Robust recursive estimation in the presence of heavy-tailed observation noise," *Ann. Statist.*, vol. 22, no. 2, pp. 1045–1080, 06 1994.

[8] S. Farahmand, G. B. Giannakis, and D. Angelosante, "Doubly robust smoothing of dynamical processes via outlier sparsity constraints," *Trans. Sig. Proc.*, vol. 59, no. 10, pp. 4529–4543, Oct. 2011.



(a) Quadrotor under attack

(b) Quadrotor under attack - IMHOTEP-SMT

Fig. 1. Performance of the quadrotor under sensor attack: (a) without secure state reconstruction algorithm, and (b) with IMHOTEP-SMT. For each scenario, we show the quadrotor trajectory (top), the estimation error on the quadrotor vertical velocity (middle), and the attack signal (bottom).

[9] Y. Mo and B. Sinopoli, "Secure estimation in the presence of integrity attacks," *Automatic Control, IEEE Transactions on*, vol. 60, no. 99, pp. 1145–1151, 2014.

[10] C.-Z. Bai and V. Gupta, "On Kalman filtering in the presence of a compromised sensor: Fundamental performance bounds," in *American Control Conference (ACC), 2014*, June 2014, pp. 3029–3034.

[11] F. Pasqualetti, F. Dorfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, Nov 2013.

[12] Y. Shoukry and P. Tabuada, "Event-Triggered State Observers for Sparse Sensor Noise/Attacks," *ArXiv e-prints*, Sept. 2013. [Online]. Available: http://arxiv.org/abs/1309.3511

[13] Y. Shoukry and P. Tabuada, "Event-triggered projected Luenberger observer for linear systems under sensor attacks," in *IEEE 53rd Annual Conference on Decision and Control (CDC)*, Dec. 2014, pp. 3548–3553.

[14] M. Pajic, J. Weimer, N. Bezzo, P. Tabuada, O. Sokolsky, I. Lee, and G. Pappas, "Robustness of attack-resilient state estimators," in *ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)*, April 2014, pp. 163–174.

[15] Y. Shoukry, A. Puggelli, P. Nuzzo, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and P. Tabuada, "Sound and complete state estimation for linear dynamical systems under sensor attack using satisfiability modulo theory solving," in *Proc. IEEE American Control conference*, 2015, pp. 3818–3823.

[16] Y. Shoukry, P. Nuzzo, A. Puggelli, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and P. Tabuada, "Secure State Estimation Under Sensor Attacks: A Satisfiability Modulo Theory Approach," *ArXiv e-prints*, Dec. 2014, [online] http://adsabs.harvard.edu/abs/2014arXiv1412.4324S.

[17] M. S. Chong, M. Wakaiki, and J. P. Hespanha, "Observability of linear systems under adversarial attacks," in *Proc. IEEE American Control conference*, 2015, pp. 2439–2444.

[18] J. Hendrickx, K. Johansson, R. Jungers, H. Sandberg, and K. C. Sou, "Efficient computations of a security index for false data attacks in power networks," *Automatic Control, IEEE Transactions on*, vol. 59, no. 12, pp. 3194–3208, Dec 2014.

[19] M. Fliess, J. Lvine, and P. Rouchon, "Flatness and defect of nonlinear systems: Introductory theory and examples," *International Journal of Control*, vol. 61, pp. 1327–1361, 1995.

[20] C. Barrett, R. Sebastiani, S. A. Seshia, and C. Tinelli, *Satisfiability Modulo Theories, Chapter in Handbook of Satisfiability*. IOS Press, 2009.

[21] Y. Shoukry, P. Nuzzo, A. Puggelli, A. L. Sangiovanni-Vincentelli, S. A. Seshia, M. Srivastava, and P. Tabuada, "IMHOTEP-SMT: A Satisfiability Modulo Theory Solver For Secure State Estimation," in *13th International Workshop on Satisfiability Modulo Theories (SMT)*, July 2015.

[22] S. Malik and L. Zhang, "Boolean satisfiability from theoretical hardness to practical success," *Commun. ACM*, vol. 52, no. 8, pp. 76–82, Aug. 2009. [Online]. Available: http://doi.acm.org/10.1145/1536616.1536637

[23] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The GRASP multiple micro-UAV testbed," *IEEE Robotics & Automation Magazine*, vol. 17, no. 3, pp. 56–65, 2010.

[24] S. Lupashin, A. Schollig, M. Sherback, and R. D'Andrea, "A simple learning strategy for high-speed quadrocopter multi-flips," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 1642–1648.