Efficient Predictive Monitoring of Linear Time-Invariant Systems Under Stealthy Attacks

Mazen Azzam, Liliana Pasquale, Gregory Provan, and Bashar Nuseibeh

***Abstract*— Attacks on industrial control systems (ICSs) can lead to significant physical damage. While off-line safety and security assessments can provide insight into vulnerable system components, they may not account for stealthy attacks designed to evade anomaly detectors during long operational transients. In this article, we propose a predictive online monitoring approach to check the safety of the system under potential stealthy false data injection attacks (FDIAs) on sensors. Specifically, we adapt previous results in reachability analysis for attack impact assessment to provide an efficient algorithm for online safety monitoring for linear time-invariant (LTI) systems. The proposed approach relies on an off-line computation of symbolic reachable sets in terms of the estimated physical state of the system. These sets are then instantiated online, and safety checks are performed by leveraging ideas from ellipsoidal calculus. We illustrate and evaluate our approach using the Tennessee–Eastman process. We also compare our approach with the baseline monitoring approaches proposed in previous work and assess its efficiency and scalability. Our evaluation results demonstrate that our approach can predict in a timely manner if an FDIA will be able to cause damage while remaining undetected. Thus, our approach can be used to provide operators with real-time early warnings about stealthy attacks.**

***Index Terms*— Control system security, ellipsoids, industrial control, linear systems, reachability analysis.**

I. Introduction

INDUSTRIAL control systems (ICSs) denote systems where safety-critical physical processes are augmented with computation and communication capabilities, e.g., transportation systems, manufacturing, and chemical processes. Recently, the security of ICS has received increasing attention, especially with the rise in the number of attacks against these systems, e.g., Ukrainian power grid blackout [30]. Different from attacks targeting IT systems, attacks against ICS can also cause physical damage, rather than only harming digital assets, e.g., sensitive data. In particular, stealthy attacks, where resourceful attackers exploit noise [4] or control theoretic properties [38] to avoid detection, can cause significant damage. Although a variety of techniques [15] consider the behavior of the physical process to detect attacks on ICS, the detection of stealthy attacks still presents several limitations [16].

Assessing the risk of stealthy attacks involves performing off-line impact assessment [6], [7], [8], [9], [10], [11], which may provide operators with more insight into potential vulnerabilities, such as the inability of a residual-based anomaly detector to detect certain sensor attacks before they cause damage. However, off-line impact assessment cannot account for potential transients and variations in operating modes that a physical system may experience. In particular, chemical plants often experience long transients and frequent changes in operating conditions due to potential unforeseen disturbances, real-time optimization modules, or high-level control decisions [28], [34]. Safety analysis consists of checking whether, from a current state, the system can enter an unsafe state given the current control settings. Conventional control methods cannot guarantee the safety of the system given the possibility of stealthy attacks to exploit noise or control-theoretic properties to avoid detection [25]. Therefore, there is a need for safety monitoring techniques that evaluate the safety of the system in real-time given such threats.

The objective of this article is to develop an efficient online safety monitoring algorithm, specifically under stealthy false data injection attacks (FDIAs) on sensors. We particularly consider a practical stealthy attack model where the attacker evades detection by ensuring that the false alarm rate of the anomaly detector is maintained [36]. To the best of our knowledge, only a few works [14], [15], [16], [17] fall into this line of research, and they either do not consider intelligently crafted stealthy attacks or are resource-intensive. Conversely, our approach to online safety monitoring provides a computationally efficient online mechanism to detect the potential impact of a range of stealthy attacks, which would be undetectable using traditional monitoring approaches. In terms of efficiency and scalability, the main feature of our approach is to perform the most computationally intensive operations offline and reduce the online safety checks to computation of a distance measure. Namely, the intensive off-line computations consist of approximating symbolic reachable sets of states. When deployed online, these sets only need to be instantiated depending on the current state estimate and a prediction of the state over a certain time horizon. We then take advantage of the geometric representation of the reachable sets to perform efficient safety checks. This general approach is inspired by

Manuscript received 16 December 2021; revised 13 May 2022; accepted 27 July 2022. Date of publication 16 August 2022; date of current version 23 February 2023. This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) and the Science Foundation Ireland (SFI) under Grant 13/RC/2094_P2 and Grant 16/RC/3918. Recommended by Associate Editor N. Quijano. *(Corresponding author: Mazen Azzam.)*

Mazen Azzam and Bashar Nuseibeh are with Lero, The Irish Software Research Centre, University of Limerick, V94 T9PX Limerick, Ireland (e-mail: mazen.azzam@ul.ie; bashar.nuseibeh@ul.ie).

Liliana Pasquale is with Lero, The Irish Software Research Centre, University College Dublin, Dublin, D04 V1W8 Ireland (e-mail: liliana.pasquale@ucd.ie).

Gregory Provan is with Lero, The Irish Software Research Centre, University College Cork, T12 K8AF Cork, Ireland (e-mail: g.provan@cs.ucc.ie).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCST.2022.3196809.

Digital Object Identifier 10.1109/TCST.2022.3196809

the work by Chen and Sankaranarayanan [9] in the context of real-time monitoring for simplex control architectures.

The main contribution of this work is an efficient and scalable online safety monitoring algorithm for linear time-invariant (LTI) systems presented with the threat of stealthy attacks. The efficiency of the algorithm stems first from the off-line precomputation of a symbolic ellipsoid approximation of the reachable set using a linear matrix inequalities (LMIs) program proposed by Murguia and Ruths [36]. Second, we take advantage of results in ellipsoidal calculus [6] and the half-space geometrical representation of unsafe sets to perform fast real-time safety checks given an instantiation of the precomputed reachable set at a predicted system state. When the algorithm is deployed online, it performs a prediction of the system state given the current state of the system and a specified time horizon. At each predicted state, the precomputed reachable set is instantiated, and emptiness checks of its intersection with the unsafe set are performed. The algorithm halts when a nonempty intersection is found or when the prediction time horizon is exhausted.

As a secondary contribution, we propose two online security metrics that can be computed by leveraging ellipsoidal calculus. The *potential impact metric* quantifies the potential impact of a stealthy attack. When the emptiness check returns a negative result, the intersection between the reachable set and the set of unsafe states can be approximated using an ellipsoid. We use the size of this ellipsoid to quantify the potential impact. When the intersection between the reachable set and the set of unsafe states is nonempty, it is also possible to compute the *time-to-unsafe metric*. This metric estimates the shortest time that an attacker would need to cause damage before being detected. This time-to-unsafe metric is fundamentally different from proximity-based metrics previously proposed in the literature [7], [8]. These metrics rely on the raw estimate of the state of the system to compute a Euclidean distance to unsafe states and are used to perform safety monitoring. Instead, our time-to-unsafe metric relies on reachable sets induced by a potential stealthy attack. As such, we account for the fact that the given estimate may not represent the real state of the system.

Finally, we evaluate the proposed algorithm using the Tennessee–Eastman process (TEP) as a case study. We first validate our algorithm through extensive simulations aimed at assessing its ability to warn about potential damage due to a stealthy attack. Second, we compare it to existing online safety monitoring techniques for attacks, namely, those that only rely on proximity-based metrics using raw state estimates. We show through simulation scenarios that, under "low-and-slow" stealthy attacks, existing techniques will not convey the security and safety situation accurately. Conversely, our reliance on reachable sets in our approach allows for early warnings to be provided to operators before a stealthy attack can cause damage. Finally, we demonstrate the suitability of the algorithm for real-time applications. Specifically, we show that safety checking takes place in a time frame that is shorter than the system's sampling period, and the algorithm scales well with the complexity of safety constraints and the desired length of time horizon for online prediction. We have applied our monitoring approach within a framework for physics-based early warnings for stealthy attacks [3].

The rest of this article is organized as follows. Section II discusses related work. Section III provides an overview of our approach. Section IV describes the adopted modeling framework. Section V details the proposed algorithm. Section VI presents numerical simulation results. Finally, Section VII concludes this article.

II. Related Work

While there exists a large body of work on model-based attack detection in control systems [15], to the best of our knowledge, approaches to tackling the problem of online safety monitoring for ICS under stealthy attacks are scarce. In model-based attack detection, we ask whether the current observations of the system are consistent with a mathematical model up to a certain degree of uncertainty. A model-based attack detection is a reactive approach to security where the occurrence of an anomaly triggers an alert. With online safety monitoring under potential attacks, we ask whether the current (estimated) state of the system can be taken by an attack on a target state that violates at least one safety constraint. Online safety monitoring is a proactive and predictive approach to safety/security that may help in guiding the selection of preemptive safety measures, such as switching to a safe and secure redundant controller [9].

Kwon and Hwang [25], [26] have proposed a recursive method to compute exact reachable sets under stealthy attacks online. While this method is computationally efficient, it uses large recursive matrices, which can make extensive use of resources. Furthermore, safety checking in this work relies on the characterization of a time-varying safe set as an ellipsoid centered at the current state. Although this is suitable for the unmanned aerial vehicle (UAV) application used by the authors, it may not be applicable in chemical process control, where unsafe operating levels are usually fixed limits imposed on physical state variables. In contrast, the bulk of the computation required for our method is performed offline, resulting in symbolic sets with a lightweight characterization when instantiated online. Furthermore, we consider more practical time-invariant unsafe sets, which can be interpreted geometrically as a union of half-spaces.

Existing online monitoring schemes [15], [16], [17], [20] rely on a notion of proximity to a predefined set of unsafe/critical states. This line of work does not consider formal safety guarantees, but it uses metrics reflecting the proximity of the system to unsafe states as a way to either determine the level of safety or detect attacks. For example, Coletta [15] and Carcano *et al.* [20] compute the minimum Euclidean distance from current states to the unsafe operating region. Castellanos and Zhou [8] extend this notion further by computing an approximate "time-to-critical-states" metric. However, these approaches rely only on raw sensor values and do not consider the effect of stealthy attacks. For example, intelligently crafted sensor attacks introduce "low-and-slow" modifications to sensor values, which may eventually not reflect the real state of the system. In our work, instead of using

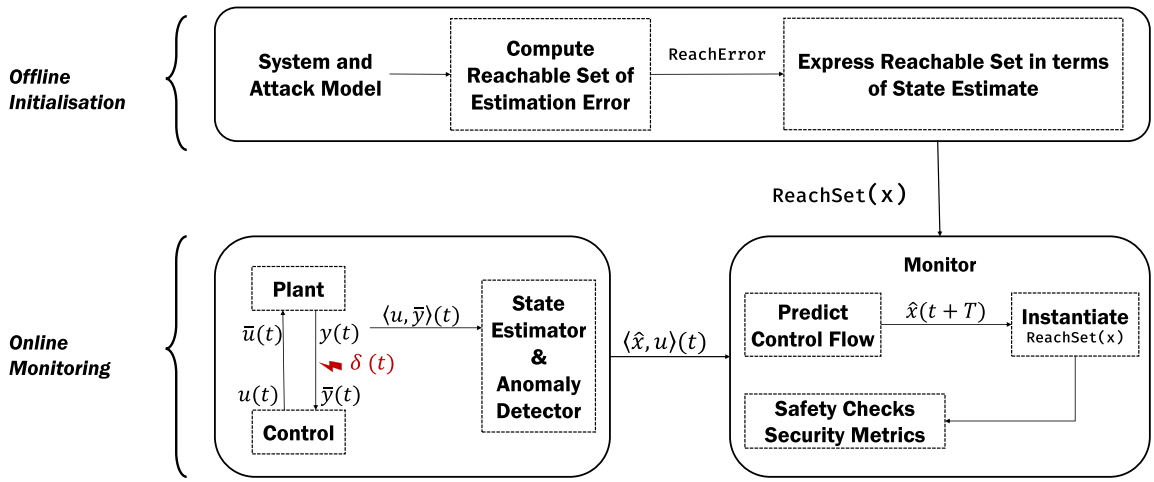

Fig. 1. Outline of the proposed online monitoring approach.

raw sensor values, we rely on reachable sets under stealthy attacks which bound—with a certain confidence level—the actual state of the system.

Another related work [7], [8], [9], [10], [11] has proposed techniques to quantify the worst case impact of potential stealthy attacks. To the best of our knowledge, these techniques are developed with the objective of performing risk assessment offline. For example, Milosevic *et al.* [32] propose a framework for security measure allocation given certain impact and attack complexity metrics. Murguia *et al.* [37] use the volume of ellipsoidal approximations of reachable sets under stealthy attacks as a measure of impact. In our work, we quantify, in real time, the potential impact of a stealthy attack based on the size of the intersection of the reachable set with the set of unsafe states. Using this intersection, instead of the entire reachable set, gives a more precise estimate of potential impact. This is made possible by using the geometric properties of the sets' representations.

Finally, our approach is inspired by recent work on real-time reachability analysis [2], [9], [43], notably in the context of real-time monitoring for simplex control architectures [21]. Similar to the algorithms proposed in this article, a few works in this area consider precomputing reachable sets offline before instantiating them online in order to perform safety checking efficiently. However, to the best of our knowledge, previous work in this context did not consider safety checking in the presence of stealthy attacks seeking to cause damage to a safety-critical system.

III. Predictive Online Monitoring Approach

In this section, we describe the online monitoring problem tackled in this article and outline the proposed approach. We also describe the main idea behind existing work on online safety monitoring under attacks. We consider this body of work as a baseline against which we compare our approach.

A. Online Monitoring Problem

This article considers the online monitoring framework depicted in Fig. 1. The objective of our approach is to check in real-time whether a potential undetected attack can cause damage to the system before being detected. In other words, given the current physical state estimate $\hat{x}(t)$, the online monitoring problem asks whether there exists a stealthy FDIA on sensors that can bring the system into an unsafe state over the next T time instants.

If this check returns a negative result, then operators can be reassured that, even if an intrusion is present, the alleged attacker may not be able to cause any damage without being detected. Otherwise, the check can serve as an early warning and can prompt operators to take preemptive safety or security measures. Such additional measures are, however, beyond the scope of this work.

B. Outline of the Proposed Approach

The proposed approach, as shown in Fig. 1, is composed of two main steps.

1) *Off-Line Initialization:* This step consists of computing symbolic reachable sets under a stealthy FDIA in terms of the state estimate based on a model of the system and attack. This is possible by considering the evolution of the state estimation error under a stealthy attack instead of the physical state itself. As a result, we express the reachable set of errors in terms of the actual state estimate. This allows us to perform the bulk of the computation offline, leading to more efficient real-time safety checking.
2) *Online Monitoring:* At runtime, the proposed monitor takes as input the current physical state estimate and the state of the controller, and predicts the value of the state up to T time steps into the future. We assume that this prediction can be done using an identified physical model of the system, with T chosen to maintain an acceptable degradation in the confidence level of the predicted state. We then instantiate the precomputed symbolic reachable set at each predicted state value and perform an emptiness check of its intersection with a predefined set of unsafe states. The prediction stops when a nonempty intersection is encountered or when

T is exhausted. Furthermore, we compute two security metrics when the intersection is nonempty:

a) *potential impact* of the attack;
b) *time-to-unsafe states* reflecting the approximate time that a potential attack would need to cause damage.

These metrics are aimed at providing operators with a better assessment of the current safety/security situation.

In this article, we apply this general approach to LTI systems, where we propose the use of ellipsoids as over-approximations of the symbolic reachable sets. Ellipsoids have been extensively used for safety verification for control systems [20], [23]. They feature an efficient quadratic representation in terms of the dimension of the state of the system [29], which presents an advantage in real-time monitoring. Furthermore, in most practical applications of process control, unsafe operating regions can be represented as unions of half-spaces. With reachable sets represented as ellipsoids, safety checking reduces to checking the sign of the distance from the ellipsoid to each of the hyperplanes composing the unsafe set [6], [23]. As a result, real-time safety monitoring is enabled with minimal resource utilization.

In our application to LTI systems, we use results by Murguia [36] in reachability analysis under stealthy attacks to precompute a symbolic reachable set in the off-line initialization phase. We then use results in ellipsoidal calculus [6], [23] to design an efficient and scalable online safety monitoring algorithm for stealthy attacks.

C. Existing Monitoring Techniques

To the best of our knowledge, existing online safety monitoring techniques [15], [16], [17], [20] rely mainly on a proximity metric to assess the safety of the system against attacks targeting physical processes. In this article, we compare our approach to these techniques that all feature the same idea detailed in the following. We provide a comparison based on intuition in this section.

Given a set of unsafe states $\mathcal{S}_u$ and the current estimated state $\hat{x}(t)$, existing online monitoring techniques [15], [16], [17], [20] compute a distance metric $d_u = \min d(\hat{x}(t), \mathcal{S}_u)$. The most commonly used distance metric is the Euclidean distance, which is mainly suitable for continuous or hybrid systems where state variables of interest assume real values. Given a set of thresholds $\tau_1 > \tau_2 > \tau_3, \ldots$, alarms of different levels of criticality can be raised based on the value of d_u. The proximity metric can further account for the dynamics of the system by computing a *time-to-unsafe/critical states* metric, $t_u = d_u/r$, where r is the approximate rate of change of the system state over a given time period [8]. In the rest of this article, the term "traditional time-to-unsafe metric" refers to t_u computed using the aforementioned formula.

However, under a stealthy attack that slowly drives the system into unsafe states to avoid detection, the real state of the system $x(t)$ will diverge from the estimated state. Thus, the metric d_u may not provide an accurate measure of the proximity of the system to unsafe states. Instead of relying merely on a proximity measure based on $\hat{x}(t)$, our online monitoring approach accounts for the possibility of stealthy attacks by considering reachable sets under such attacks. Namely, if $\mathcal{R}_x(t)$ is the reachable set of states under a stealthy attack given the used anomaly detector at time t, then $x(t) \in \mathcal{R}_x(t)$ even if $x(t) \neq \hat{x}(t)$. Otherwise, by definition, the attack would be detected by the anomaly detector. If, over the next T time period, $\mathcal{R}_x(t') \cap \mathcal{S}_u \neq \emptyset$ for some $t' \in [t; t+T]$, then it is possible for an attacker to drive the system into the unsafe operating region within at least $t' - t$ time units. As such, our safety monitoring approach relies on the emptiness checking of this intersection, instead of a mere proximity measure. Furthermore, we consider the proximity metric $t' - t$ as an additional security metric that may assist operators in assessing the current safety/security situation.

IV. Modeling Framework

We apply our approach in this article to LTI systems. Before detailing the proposed approach, we briefly describe the layout of the system and the attack model in this section.

A. System Layout

Consider the control system architecture in Fig. 1. We assume that the physical system can be approximated using an LTI model

$$\begin{cases} x(k+1) = Ax(k) + B\bar{u}(k) + w(k) \\ y(k) = Cx(k) + v(k) \end{cases} \tag{1}$$

where $x(k) \in \mathbb{R}^n$ is the state vector, $\bar{u}(k) \in \mathbb{R}^l$ denotes the control signals received by the system, and $y(k) \in \mathbb{R}^m$ is a vector of sensor measurements. $w(k) \in \mathbb{R}^n$ and $v(k) \in \mathbb{R}^m$ denote process and measurement noise, respectively, and are assumed to follow a zero-mean Gaussian distribution with respective covariance matrices Σ_1 and Σ_2. $k = t/\Delta_t \in \mathbb{N}$ denotes the discrete time instants, where Δ_t is the sampling period and t is the continuous time. A, B, and C are real, time-invariant matrices of appropriate dimensions. The system is assumed to be equipped with an output feedback control loop such that $u(k) = \mathcal{K}(\bar{y} - y_r(k))$, where $u(k)$ denotes control signals that are originally sent to the process, $\mathcal{K}$ is the control law, $y_r(k)$ denotes the reference output, and $\bar{y}(k)$ denotes measurements received by the controller. In this work, we focus on sensor attacks such that $u(k) = \bar{u}(k)\ \forall k$.

Furthermore, we assume that a subset of state variables, denoted as critical, is grouped in the vector $x_c = C_c x$, $x_c \in \mathbb{R}^{n_c}$, $C_c \in \mathbb{R}^{n_c \times n}$ and is required to remain within a certain safe set to ensure safe operation. Let $\mathcal{S}_u$ be the set of unsafe states; we assume that unsafe conditions are given as a linear combination of the critical state variables such that the unsafe set becomes a union of half-spaces[1]

$$\mathcal{S}_u = \left\{ x(k) \in \mathbb{R}^n \;\middle|\; \bigcup_{i=0}^{n_c} C_{c,i} x_i(k) \geq b_i \right\} \tag{2}$$

where $b_i \in \mathbb{R}$ is the ith half-space scalar, $C_{c,i}$ denotes the ith row of the matrix C_c, and $x_i(k)$ denotes the ith element of $x(k)$.

[1]This assumption is typical in several process control applications.

At a time k, given previous sensor measurements and control actions, a Kalman filter generates an estimate of the physical state and expected sensor measurements as follows:

$$\begin{cases} \hat{x}(k) = A\hat{x}(k-1) + Bu(k-1) \\ \qquad\quad + L(\bar{y}(k-1) - C\hat{x}(k-1)) \\ \hat{y}(k) = C\hat{x}(k) \end{cases} \tag{3}$$

where L denotes the Kalman gain, and $\hat{x}$ and $\hat{y}$ denote estimated state and measurements, respectively.

In addition, a chi-squared anomaly detector compares received measurements with the generated estimate through a residual $r(k) := \bar{y}(k) - \hat{y}(k)$. Under nominal conditions, the residual metric has a zero-mean and a covariance matrix Σ. To check for this hypothesis, a chi-squared metric, $z(k) = r^T(k)\Sigma^{-1}r(k)$, is computed and compared with a threshold τ, such that exceeding this threshold implies a possible anomaly and raises an alarm. The threshold τ is designed to maintain a certain false alarm rate β such that $\Pr[z(k) \leq \beta] = 1 - \beta$ under the nominal operation and can be set as described by Murgia *et al.* [36] (see Proposition 1).

B. Attacker Model

We consider in this work FDIAs on sensors that are masked by the system noise in order to drive the latter slowly to the unsafe set (2). While different stealthy attack strategies exist in the literature, we choose to focus on one that is feasible in practice. Hence, we provide the following justification for our particular choice as opposed to other possible stealthy attacks.

1) We consider FDIAs specifically on sensors as it has been shown both theoretically [27] and empirically [45] that their stealthiness is easier to maintain than attacks on actuators. FDIAs that consist of corrupting all sensor and actuator channels, also known as "covert" attacks [41], even though undetectable, require a significant amount of resources to be executed [42]. In practice, it may not be possible for an attacker to have simultaneous access to all sensor/actuator communication channels or associated control devices. Each sensor/actuator may require different kinds of software/hardware tools in order to corrupt the data they send or receive [1].
2) Replay attacks, similar to FDIAs, threaten the integrity of sensor measurements. Although they require fewer resources to remain stealthy [42], the stealthiness of replaying old sensor measurements relies on whether these measurements could be admissible at a given time. Therefore, the system needs to be operating in a steady state for the replayed measurements to be considered nominal by the anomaly detector. If transients are experienced, then replaying old measurements would reveal the attack since they no longer correspond to the current control inputs. In this work, we consider process control systems whereby long transients may be experienced, thus limiting the extent to which stealthy replay attacks may be successful.

In our attack model, we assume that the attacker has sufficient resources and knowledge about the system, including knowledge of the system dynamics and anomaly detector properties. Let $\{k_s, \ldots, k_f\}$ denote the time period of the attack; we model the attack as a bias imposed on sensor measurements

$$\bar{y}(k) := y(k) + \delta(k) \quad \forall\, k \in \{k_s, \ldots, k_f\}. \tag{4}$$

The attack $\delta(k)$ remains stealthy by ensuring that the false alarm rate is maintained throughout the attack period. We use this characterization because anomaly detectors do raise alarms under nominal operation. A sudden disappearance of these alarms in practice may raise suspicion in operators and lead them to uncover the attack before it can cause damage [18]. Under the attack in (4), the residual is given by $r(k) = \bar{y}(k) - \hat{y}(k) = y(k) - \hat{y}k + \delta(k)$. As such, the chi-squared metric under attack is given by

$$z(k) = (y(k) - \hat{y}(k) + \delta(k))\Sigma^{-1}(y(k) - \hat{y}(k) + \delta(k)). \tag{5}$$

By selecting $\delta(k)$ to be such that $\Pr[z(k) \leq \tau] = 1 - \beta$, the attacker manages to remain undetected. For example, given K time steps, the attacker may choose to raise alarms for βK steps so that the false alarm rate is mimicked as closely as possible [18].[2] This is possible since we assume that the attacker knows the detector's parameters, i.e., Σ, β, and τ, in addition to the system and estimator outputs, i.e., $y(k)$ and $\hat{y}(k)$, respectively. In practice, such information could be obtained, for example, through reconnaissance attacks or insider knowledge. In addition, while this attack strategy is specifically designed for chi-squared detectors, most existing work in model-based anomaly detection employs this statistical change detection test [35].

We note that optimal stealthy FDIA strategies that take advantage of system noise are studied in the literature [37], [38], [39]. However, our choice for the previous attack model stems from its relative simplicity and practicality. In works describing optimal stealthy attacks, the attacker requires a deeper knowledge and understanding of the dynamics of the system. With the adversary model that we adopt, the attacker is required to only know the basic parameters of the system model and its anomaly detector, which are relatively easily obtained through a reconnaissance phase. Furthermore, optimizing the attack strategy comes at a significant computational cost for the attacker and, thus, limits the attack's real-time performance, especially if the attacker needs to adapt to an adaptive defender. We also note that our approach requires only the attack to mimic the false alarm rate of the anomaly detector. As such, it can be applied to optimized attack strategies that also take advantage of the false alarm rate to remain stealthy.

V. Proposed Monitoring Algorithm

The proposed approach relies mainly on the off-line computation of the symbolic reachable set of estimation errors under the attack described in (4). This set is an ellipsoidal overapproximation of the exact reachable set, parameterized by the state estimate. In real time, given the state estimate at time k, the symbolic set is instantiated at the K-step predicted

[2]Due to space limitations, the reader is referred to [18] for a more detailed description of distribution of the detector metric under such attack.

state. Emptiness checks of its intersection with $\mathcal{S}_u$ are then performed. We detail both the off-line and online computations in the following, and we propose online security metrics based on the computed reachable set.

A. Off-Line Computation of Symbolic Reachable Set

To compute the reachable set of the estimation error under the attack in (4), we use the method described by Murgia and Ruths [36]. We define this error as $e(k) := x(k) - \hat{x}(k)$ and assume that, at the start of an attack, the estimation error is always almost zero. The reachable set of the estimation error under the attack in (4) is independent of the actual physical state at the start of the attack. As such, this set serves as a symbolic reachable set parameterized by the state estimate.

By setting $e(k) = x(k) - \hat{x}(k)$, and performing some algebraic manipulations of (3), the evolution of the estimation error under an attack is given by

$$\begin{aligned} e(k+1) &= Ae(k) - L(Ce(k) + v(k) + \delta(k)) + w(k) \\ &= Ae(k) - L(y(k) - \hat{y}(k) + \delta(k)) + w(k). \end{aligned} \quad (6)$$

Since the error is partially driven by the Gaussian noise $w(k)$ and the attack-dependent sequence $\bar{\delta}(k) = y(k) - \hat{y}(k) + \delta(k)$, using a deterministic approach will yield an unbounded reachable set, as the support of $w(k)$ and $\delta(k)$ [as characterized in (4) and (5)] is infinite. This issue can be overcome by setting a confidence level on the energy of both of these vectors. For the attack, the sequence $\bar{\delta}(k)$ is already constrained to be such that $\Pr[z(k) \leq \tau] = \Pr[\|\Sigma^{-1/2}\bar{\delta}(k)\|^2 \leq \tau] = 1 - \beta$, where $\|\cdot\|$ denotes the L_2-norm. For the noise, let $p = \Pr[\|w(k)\|^2 \leq \bar{w}]$; since $w(k)$ follows a zero-mean Gaussian distribution, the bound $\bar{w}$ on $\|w(k)\|^2$ can be determined using the gamma distribution for a desired confidence p.

By using this assumption, the resulting reachable set can be interpreted as a level set of the distribution of the reachable error. A larger confidence level would lead to a larger set at the cost of being overly conservative with the safety checking. A reasonable choice for p would be $1-\beta$, as the false alarm β is designed to be small. This also simplifies the computation of the reachable set since, for $p = 1 - \beta$, we readily have $\Pr[\|w(k)\|^2 \leq \bar{w}] = \Pr[z(k) \leq \tau]$ under the attack in (4). The following is based on this choice; for a more detailed treatment of this confidence level and a comparison of reachable sets under different choices of p, the reader is referred to [36] and [18] due to space limitations. Let $\mathcal{R}_e^p$ denote the reachable set of error under the attack in (4) and a confidence level $p = 1 - \beta$

$$\begin{aligned} \mathcal{R}_e^p := \{e(k) \in \mathbb{R}^n \mid e(k) \text{ is s.t. (6)}, \\ p = \Pr[\|w(k)\|^2 \leq \bar{w}] = 1 - \beta\}. \end{aligned} \quad (7)$$

While computing $\mathcal{R}_e^p$ is intractable, it is possible to overapproximate the set using an ellipsoid in $\mathbb{R}^n$, given by

$$\mathcal{R}_e^p \subseteq \mathcal{E}_e^p = \{e(k) \mid e^T(k)\Pi^{-1}e(k) \leq 1\} \quad (8)$$

where the positive definite matrix Π is the ellipsoid's shape matrix. Letting $\mathcal{P} = \Pi^{-1}$, the minimum volume ellipsoid containing the set $\mathcal{R}_e^p$ can be obtained by solving the following semidefinite program [36]:

$$\begin{aligned} \mathcal{P} = \arg\min\ & -\log\det\mathcal{P} \\ \text{s.t. } & \mathcal{P} > 0; \quad \mathcal{Q} \geq \mathbf{0} \end{aligned} \quad (9)$$

where

$$\mathcal{Q} = \begin{bmatrix} b\mathcal{P} & A^T\mathcal{P} & \mathbf{0} & \mathbf{0} \\ \mathcal{P}A & \mathcal{P} & \mathcal{P} & -\mathcal{P}L\Sigma^{1/2} \\ \mathbf{0} & \mathcal{P} & \dfrac{1-b}{\tau+\bar{w}}I & \mathbf{0} \\ \mathbf{0} & -\Sigma^{1/2}L^T\mathcal{P} & \mathbf{0} & \dfrac{1-b}{\tau+\bar{w}}I \end{bmatrix} \quad b \in (0,1). \quad (10)$$

Note that, while b is an optimization variable, it is necessary to fix it to ensure the convexity of the program. A grid search can then be performed over the interval $(0, 1)$ to find the optimal shape matrix corresponding to the minimum-volume ellipsoid.

Given the shape matrix $\Pi = \mathcal{P}^{-1}$, and replacing $e(k)$ by its definition, we obtain a symbolic ellipsoidal approximation $\mathcal{E}_x^p(k)$ of the reachable set $\mathcal{R}_x^p(k)$ of the actual system state $x(k)$, parameterized by the current state estimate $\hat{x}(k)$

$$\begin{aligned} \mathcal{R}_x^p(k) \subseteq \mathcal{E}_x^p(k) = \{x(k) \in \mathbb{R}^n \mid (x(k) - \hat{x}(k))^T \\ \times \Pi^{-1}(x(k) - \hat{x}(k)) \leq 1\}. \end{aligned} \quad (11)$$

Algorithm 1 summarizes the off-line steps to obtain $\mathcal{E}_x^p(k)$. Given the system matrix A, the Kalman gain L, the residual covariance matrix Σ, the anomaly detector's threshold τ, and the confidence bound $\bar{w}$, the algorithm performs a grid search over $(0, 1)$ by partitioning the interval into segments of length Δh. The choice of Δh will depend on the desired tightness of the ellipsoidal approximation given the computational resources available. Note that this step only needs to be performed offline once, and only the matrix Π needs to be stored to instantiate $\mathcal{E}_x^p(k)$ online given a state estimate $\hat{x}(k)$.

Algorithm 1 Off-Line Symbolic Reachable Set Computation

INPUTS: $(A, L, \Sigma, \tau, \bar{w}, \Delta h)$; $0 < \Delta h < 1$
OUTPUT: Reachable set shape matrix $\mathbf{\Pi}$

$b \leftarrow \Delta h$;
SolutionList $\leftarrow$*EmptyList*();
while $b < 1$ **do**
▷ Solve the programme in (9) for the current value of b
  SolveSemiDefiniteProgramme$(A, L, \Sigma, \tau, \bar{w}, b)$;
  SolutionList.*append*(CurrentSolution);
  $b \leftarrow b + \Delta h$;
end while
BestShapeMatrix $\leftarrow$ *MinObjectiveValue*(SolutionList);
return BestShapeMatrix;

B. Online Safety Checks

Algorithm 2 outlines the steps needed to perform online safety checks. Given the current state estimate $\hat{x}(k)$ and the state of the controller, we estimate the state of the system for

Algorithm 2 Online Safety Checking

INPUTS: $(K, \mathbf{\Pi}, \hat{x}(k)$, ControllerState, UnsafeSet)
OUTPUT: *true* if the system is safe under a potential stealthy attack; *false* otherwise

$\hat{x}_p \leftarrow \hat{x}(k)$
for all $l \in \{0, 1, \ldots, K\}$ **do**
  ReachEll $\leftarrow$*Ellipsoid*($\hat{x}_p$,$\mathbf{\Pi}$);
  for all Hyperplane $\subset$ UnsafeSet **do**
    DistToUnsafe $\leftarrow$ *dist*(ReachEll,Hyperplane);
    if DistToUnsafe ≤ 0 **then**
      return *false*;
    end if
  end for
  $\hat{x}_p \leftarrow$ *PredictControlFlow*($\hat{x}_p$,ControlState);
end for
return *true*;

K time steps into the future using the identified model of the plant. At each time step $l \in \{0, \ldots, K\}$, we instantiate $\mathcal{E}_x^p(k+l)$, and we check whether it intersects the set $\mathcal{S}_u$. The algorithm halts and reports an unsafe state when a nonempty intersection is encountered. If the prediction horizon is exhausted, the algorithm reports a safe state. In the following, we detail the procedure that we use to perform the emptiness checks.

Let $\mathcal{H}_i = \{x \in \mathbb{R}^n \mid C_{c,i}x \geq b_i\}$ be a half-space representing one of the safety conditions composing the set $\mathcal{S}_u$ [see (2)]. Checking whether $\mathcal{E}_x^p(k_f) \cap \mathcal{S}_u = \emptyset$ involves checking whether $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i = \emptyset$ for each $i \in \{1, \ldots, n_c\}$. If the latter is true for all i, then the former is also true since $\mathcal{S}_u = \cup_{i=1}^{n_s} \mathcal{H}_i$.

To check whether $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i = \emptyset$, it suffices to compute the minimum distance from $\mathcal{E}_x^p(k_f)$ to the hyperplane that delimits the half-space $\mathcal{H}_i$. Let $\mathcal{H}_{p,i} = \{x \mid C_{c,i}x = b_i\}$ be such hyperplane; the minimum distance from $\mathcal{E}_x^p(k_f)$ to $\mathcal{H}_{p,i}$ is given by [24]

$$d_i(k_f) = \frac{|b_i - C_{c,i}x(k_f)| - \sqrt{x(k_f)^T \Pi x(k_f)}}{\left\| C_{c,i}^T \right\|}. \tag{12}$$

If $d_i(k_f) \leq 0$, then $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i \neq \emptyset$. Otherwise, if $d_i(k_f) \geq 0$, then the ellipsoid $\mathcal{E}_x^p(k_f)$ is either contained in $\mathcal{H}_i$ or does not intersect the half-space, depending on whether its center $\hat{x}(k_f)$ belongs to $\mathcal{H}_i$. However, since the state estimate is within the safe region,[3] i.e., $\hat{x}(k_f) \notin \mathcal{H}_i$, then, in our case, $d_i(k_f) > 0$ always implies that $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i = \emptyset$.

C. Real-Time Security Metrics

In addition to checking the emptiness of the intersection of the reachable set with the set of unsafe states, it is possible to derive two online security metrics. The first metric can help operators get a better idea of the potential impact of a stealthy false data-injection attack, while the second approximates the minimum amount of time that would be required for an attacker to cause damage. In this section, we show how ellipsoidal methods can be used to compute such metrics efficiently.

1) Real-Time Impact of Stealthy Attack: In the case where $\mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset$, we can quantify the impact of a potential stealthy false data-injection attack using the approximate size of this intersection. Namely, for each half-space $\mathcal{H}_i \subset \mathcal{S}_u$, it is possible to overapproximate $\mathcal{E}_x^p(k_f) \cap \mathcal{H}_i$ using a minimum-volume ellipsoid $\mathcal{E}_i(k_f)$ of center $q_i(k_f) \in \mathbb{R}^n$ and shape matrix $\Pi_i(k_f)$ as follows [6]:

$$\begin{aligned} q_i(k_f) &= \hat{x}(k_f) - \frac{1+\alpha_i n}{n+1}\Pi\bar{c}_i \\ \Pi_i(k_f) &= \frac{n^2(1-\alpha_i^2)}{n^2-1} \\ &\quad \times \left(\Pi - \frac{2(1+\alpha_i n)}{(n+1)(\alpha_i+1)}\Pi\bar{c}_i\bar{c}_i^T\Pi\right) \end{aligned} \tag{13}$$

where $\bar{c}_i = C_{c,i}/(C_{c,i}\Pi C_{c,i}^T)^{0.5}$ and $\alpha_i = (C_{c,i}\hat{x}(k_f) - b_i)/(C_{c,i}\Pi C_{c,i}^T)^{0.5}$. As such, we quantify the impact of a potential stealthy false data-injection attack using the volume of $\mathcal{E}_i(k_f)$. The volume of a general ellipsoid in $\mathbb{R}^n$ with a shape matrix Q is given by

$$\mathbf{vol}(\mathcal{E}) = \mathbf{vol}[\mathcal{B}_n]\sqrt{\det Q} \tag{14}$$

where $\mathbf{vol}[\mathcal{B}_n]$ and $\det Q$ denote the volume of the unit n-ball and the determinant of the matrix Q, respectively. It is worthwhile to note that different system dimensions may lead to vastly different number ranges for the volume of the intersection ellipsoid. Thus, in order to make the impact metric more meaningful, we propose to use the ratio of the volume of the intersection ellipsoid to that of the ellipsoid approximating the reachable set. This guarantees that the impact metric will fall in the range [0; 1], thus becoming more intuitive to interpret. From (14), the impact metric reduces to the following:

$$\mathbf{Im}(k) = \left[\max_{i=1,\ldots,n_c} \det \Pi_i(k_f)\right] \Big/ \det \Pi. \tag{15}$$

2) Approximate Time to Unsafe States: In the case $\mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset$ for some $k_f \in \{k, \ldots, k+K\}$, we use the time k_f as the approximate time to unsafe states metric in our approach, namely,

$$\begin{aligned} \mathbf{Tc}(k) = (k_f - k)\Delta_t, \quad & \exists k_f \in \{k, \ldots, k+K\} \\ \text{s.t. } & \mathcal{E}_x^p(k_f) \cap \mathcal{S}_u \neq \emptyset \end{aligned} \tag{16}$$

where Δ_t is the system's sampling period. The advantage of using k_f instead of the distance from the state estimate $\hat{x}(k)$ itself is that the former approach accounts for the fact that, if an undetected attack is present, then the actual state $x(k_f) \neq \hat{x}(k_f)$ still lies within $\mathcal{E}_x^p(k_f)$ (otherwise, the attack would be detected). This metric provides operators with an idea of the minimum time that they have to react before a potential stealthy FDIA manages to bring the system into an unsafe state.

[3]Otherwise, it would be clear that the system is evolving to an unsafe state, and Algorithm 2, in this case, would become obsolete.

TABLE I
SAFETY CONSTRAINTS CONSIDERED FOR THE TE CASE STUDY [12]

Output	Low Limit	High Limit
Reactor Pressure	none	2895 kPa
Reactor Temperature	none	150 $^{\circ}C$
Reactor Level	11.8 m^3	21.3 m^3
Product Separator Level	3.3 m^3	9.0 m^3
Stripper Base Level	3.5 m^3	6.6 m^3

VI. EVALUATION

In this article, we use the TEP with the control architecture designed by Ricker [39] as a case study. This benchmark process is widely regarded as one that reflects a high degree of accuracy a real-life chemical process [22]. In addition, it has been used widely to test ideas in process control [40] and model-based approaches to ICS security [14].

Namely, we used the simulation written in Simulink by Bathelt *et al.* [5]. While the controllers are implemented as Simulink blocks, the physical process itself is simulated in continuous time and is written in C and incorporated into Simulink using MATLAB's S-function blocks. Rate transition blocks are, in turn, used to simulate the discrete-time sampling of sensor measurements and actuator signals by digital controllers, in a fashion that mimics real-life situations in process control systems. We implemented Algorithms 1 and 2 in MATLAB with the ellipsoidal techniques based on the Ellipsoidal Toolbox written by Kurzhanskiy and Varaiya [24]. We approximated the TEP process as an LTI system with 50 state variables using MATLAB's `n4sid` algorithm. We considered the safety constraints discussed by Down and Vogel [12], as shown here in Table I. We augmented the TEP Simulink simulation with a Kalman filter and a chi-squared anomaly detector. To initialize the online monitoring tool, we ran Algorithm 1 to determine the reachable ellipsoid's shape matrix. We performed the grid search for parameter b by dividing the interval into segments of length 0.01.

Our evaluation consists of three main parts. First, we validated our approach by measuring true and false positive rates using extensive simulations. Second, we compared our approach with existing online monitoring approaches. Finally, we assessed the performance and scalability of our approach.

A. Validation

The objective of our approach is not to detect attacks but rather to perform safety checking under *potential* stealthy attacks that seek to cause damage. Namely, Algorithm 2 checks whether the current state of the system can be taken to an unsafe state by a stealthy attack within the next K time instants and cause damage before the anomaly detector detects the attack. Thus, to evaluate our approach, we ran several simulations of the TEP only considering different stealthy attacks on the sensors that report values of safety-critical parameters shown in Table I. We avoided using the true/false positive/negative rate performance metrics as traditionally defined in the attack detection literature. Instead, we considered a notion of true/false positives/negatives similar to the one adopted in previous work on online safety monitoring [10].

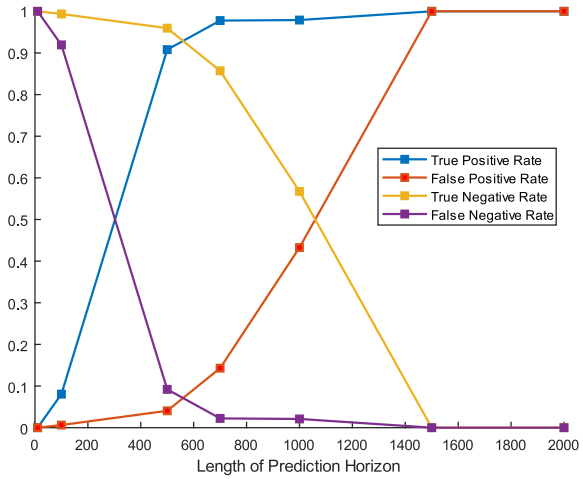

Fig. 2. True/false positive/negative rates as a function of the length of prediction horizon K.

1) A *true positive* occurs when Algorithm 2 raises a warning within K time instants before damage occurs due to an attack, and the system reaches an unsafe state before the anomaly detector raises an alarm.
2) A *true negative* occurs when Algorithm 2 does not raise any warnings within K time instants before damage occurs, but the attack is detected by the anomaly detector.
3) A *false positive* occurs when Algorithm 2 reports an unsafe state within K time instants before damage occurs, but the anomaly detector manages to detect the attack before the system enters the unsafe state.
4) A *false negative* occurs when Algorithm 2 does not raise any warnings within K time instants before damage occurs, even if a stealthy attack is taking place, and the anomaly detector does not raise any alarm.

We first tested the effect of the length of the prediction horizon K on these rates, with results presented in Fig. 2. For each value of K that we tested, we ran 500 simulations where we picked the attacked sensors at random, and we simulated the attack as a slowly growing bias on sensor measurements.

We can see from Fig. 2 that, for a small prediction horizon length, Algorithm 2 returns mostly negative checks, with true and false negatives accounting for the vast majority of predictions for $K < 500$. As K grows, the number of true and false positives increases, with the false positive rate increasing in a much slower manner. For $K \geq 1500$, although the rate of true positives is high, Algorithm 2 returns a high number of false positives as well. This behavior is the result of the design of Algorithm 2. First, for small K, the algorithm will likely not be exploring a sufficient number of states where a stealthy attack would cause a violation of safety constraints. Thus, it is expected to observe a high rate of both false and true negatives, with true and false positive rates remaining very low. As K increases, the algorithm is allowed to explore more states, therefore increasing the number of true

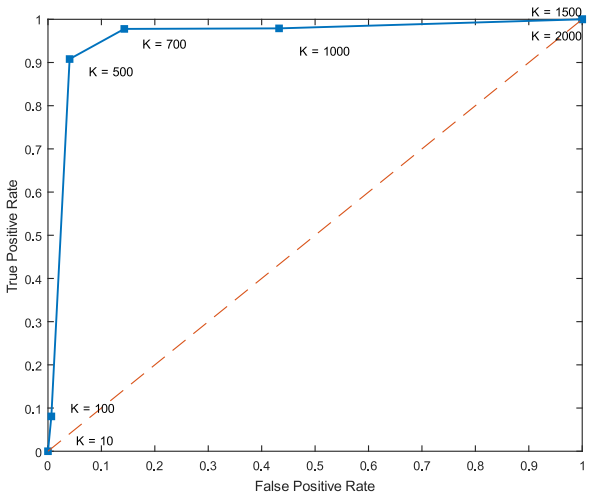

Fig. 3. ROC curve for Algorithm 2 with the length of prediction horizon K as the third dimension.

TABLE II

TRUE POSITIVE/NEGATIVE RATES VERSUS THE NUMBER OF SENSORS ATTACKED AT THE SAME TIME

Attacked Sensors	*1*	*2*	*3*	*4*	*5*
True Positive Rate	0.908	0.905	0.91	0.905	0.92
False Positive Rate	0.0405	0.041	0.04	0.0408	0.04

positives. The slow parallel increase in false positives shows that Algorithm 2 exhibits high accuracy for intermediate values of K. However, at high values of K, the accuracy of the predicted states is expected to decrease, which explains the high false positive rates.

These simulations show that there exists a tradeoff between how early we would like to raise warnings about potential safety violations due to a stealthy attack and the accuracy of Algorithm 2. These experiments can also serve as a method to tune the choice of K. To showcase these ideas, we have plotted in Fig. 3 the receiver operating characteristic (ROC) curve for Algorithm 2 with the length of prediction horizon K as the third dimension. We note that it exists a "cutoff" point at $K = 500$ time steps where we obtain acceptable values for the true/false positive rates (90.8% for true positives and 4.05% for false positives). This is equivalent to about 15 minutes ahead-of-time prediction, which is a reasonable choice in practice for K.

For $K = 500$, we tested the accuracy of Algorithm 2 under different numbers of sensors being attacked at the same time. We ran 500 simulations for each different number of sensors being attacked. In each simulation, we picked the attacked sensors at random, and we ran Algorithm 2 while considering the safety constraints associated with the sensor(s) under attack (see Table I). The results in Table II show high true positive and low false positive rates in each case. These experiments demonstrate the accuracy of Algorithm 2 with respect to all safety-critical sensors. Given the large number of random simulations that we ran, we can conclude that Algorithm 2 can report potential safety violations due to a stealthy attack with respect to all the safety constraints imposed on the system.

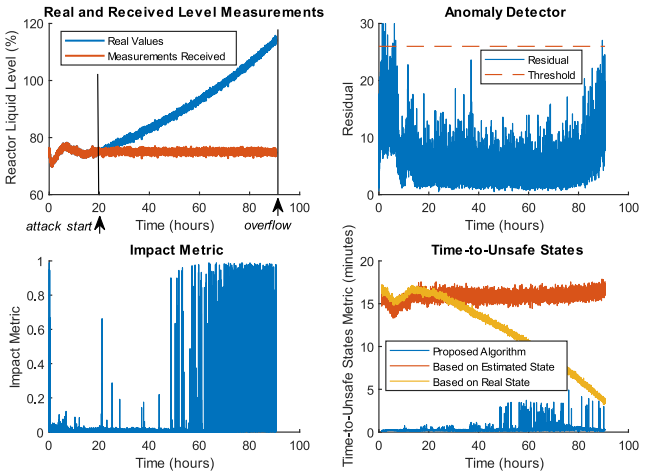

Fig. 4. Results from simulation Scenario 1.

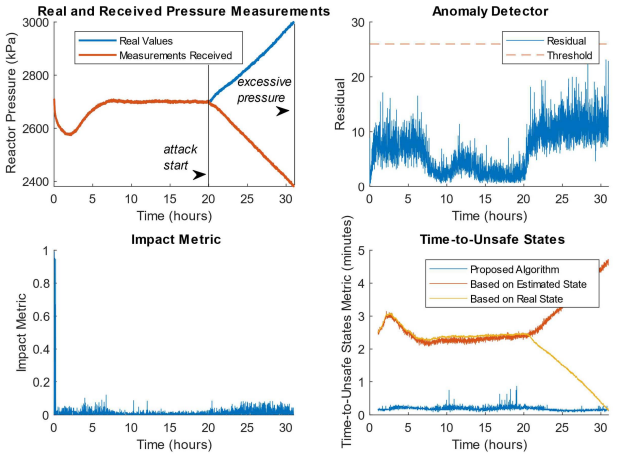

Fig. 5. Results from simulation Scenario 2.

B. Comparison With Existing Monitoring Approaches

In this section, we empirically showcase the usefulness of our approach compared to existing monitoring approaches described in Section III-C. To this end, we implemented an online monitoring tool measuring the time-to-unsafe states metric based on the Euclidean distance from the current state estimate to the set of unsafe states. We also measure the average rate of change of the estimated system state. We avoided a comparison based on the accuracy metrics depicted in Section VI-A. This is due to the fact that the traditional time-to-unsafe states metric relies on the selection of different thresholds to raise alarms of different criticality. With the lack of precise methods to select these thresholds, it is hard to perform a meaningful quantitative comparison between the metric proposed in this article and the traditional one. Therefore, we used a set of attack scenarios on safety-critical sensors to empirically demonstrate the advantages of our approach. We particularly focus on attacks targeting sensors with a slowly growing bias.

We chose three attack scenarios. Scenarios 1 and 2 depict individual attacks targeting the level and pressure sensors, respectively, of the main reactor in the TEP. Scenario 3 depicts an attack performed simultaneously on the level,

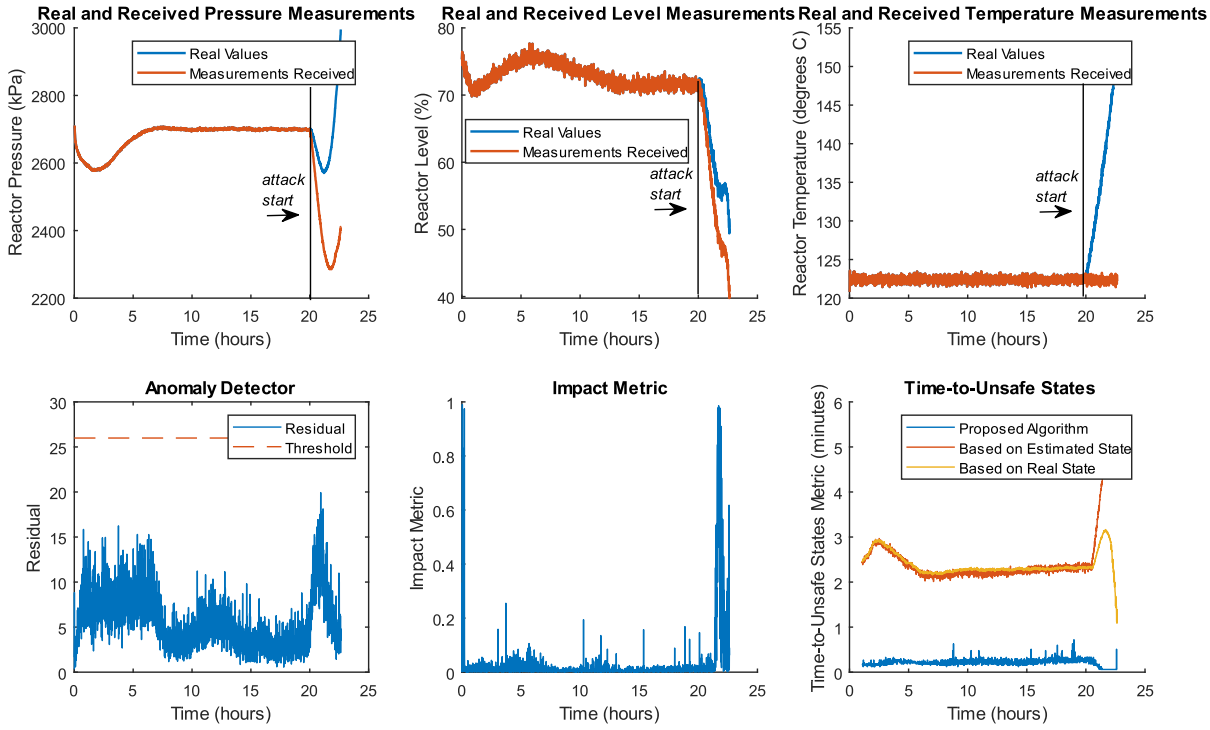

Fig. 6. Results from simulation Scenario 3.

pressure, and temperature sensors of the reactor. We chose Scenarios 1 and 2 to illustrate the typical kind of attacks targeting safety-critical sensors, individually. We can obtain similar results for individual attacks on other safety-critical sensors. Scenario 3 illustrates a more dangerous coordinated attack where all main reactor sensors in the TEP are manipulated at the same time. Again, similar results can be obtained for other combinations of sensors for safety-critical variables listed in Table I. Figs. 4–6 present the results obtained for each scenarios.

1) Scenario 1: In this scenario, we simulate an attack on the reactor's level sensor, where a growing bias on level measurements is introduced to trick the controller into overflowing the reactor. This simulation is shown in Fig. 4 where we can see that the anomaly detector raises an alarm almost at the moment the overflow takes place. Our impact metric increases significantly over the period preceding the physical damage to the reactor, and our time-to-unsafe states metric shows that the plant is dangerously close to damage. Conversely, the traditional time-to-unsafe states computed based on the estimated state alone show a slight increase, indicating that the plant appears to be moving away from the unsafe operating region.

2) Scenario 2: In this scenario, we perform an attack on the reactor's pressure sensor. Namely, the attack is a slowly growing bias on pressure measurements seeking to trick the pressure controller into increasing the pressure inside the reactor, while it appears lower than its set point. We can see from Fig. 5 that the anomaly detector fails to raise any alarm before excessive pressure builds up in the reactor. However, our online monitoring algorithm reports that the plant's operation may be unsafe in the presence of an attack throughout this ramp-down operation, which is shown by a nonzero impact metric. While the impact metric shows an increase over the few hours between the start of the attack and the damage taking place. Instead, the traditional time-to-unsafe states metric shows the plant moving away from unsafe states.

3) Scenario 3: In this scenario, we simulate simultaneous attacks on the main reactor's pressure, temperature, and level sensors. All three attacks are slowly growing biases. Fig. 6 shows that damage occurs faster in this scenario than in the previous two, with the anomaly detector again failing to raise any alarms. Our impact metric, however, shows again that the plant's operation may be unsafe under the attack. Instead, the traditional time-to-unsafe states metric depicts the plant moving away from the unsafe operating region.

These scenarios demonstrate the usefulness of our approach in the presence of stealthy attacks compared to simple distance-to-unsafe metrics. Relying on the traditional distance-to-unsafe metric may relay an inaccurate idea of the current security or safety conditions. This was especially highlighted in Scenarios 2 and 3. While the plant appears to drift away from the unsafe operating region, our monitoring approach can still warn operators that an attacker is able to damage the system without being detected.

Fig. 7 shows a comparison between the time-to-unsafe metric computed using our algorithm and the same metric computed based on the raw estimated state. Namely, we plot the difference (error) between the metric in each case and the time-to-unsafe states computed based on the real state of the system. In each scenario, we observe that the metric based on the raw estimated state is relatively accurate before the attack starts (the error is close to zero). However, the error starts to grow as the stealthy attack progresses, and the real state diverges from its estimate. Conversely, as the stealthy attack progresses, this error decreases for the time-to-unsafe

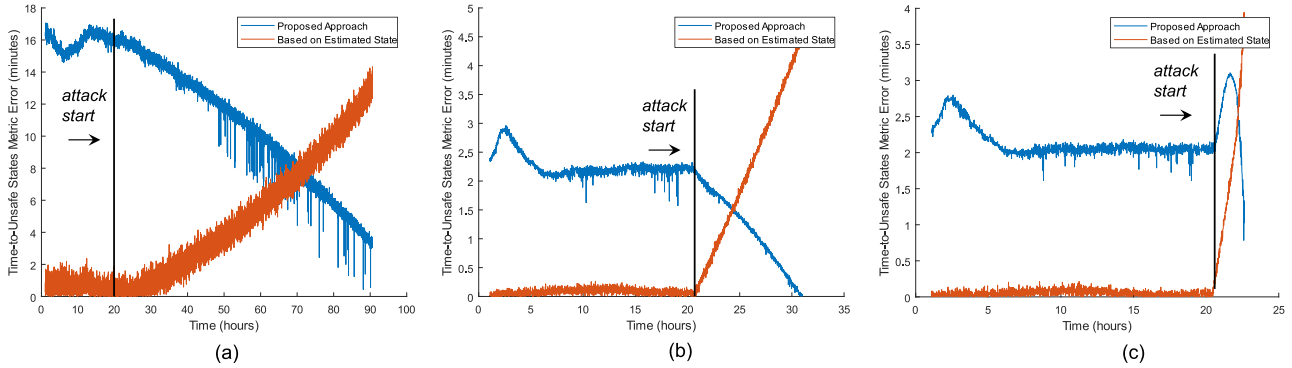

Fig. 7. Difference (error) between time-to-unsafe metric computed based on state estimate and based on the proposed algorithm versus the time-to-unsafe states based on the real state of the system. (a) Scenario 1. (b) Scenario 2. (c) Scenario 3.

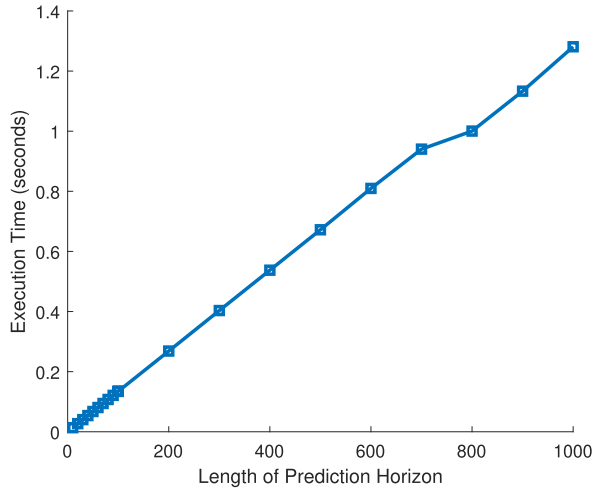

Fig. 8. Average execution time of the proposed scheme versus the number of steps for prediction.

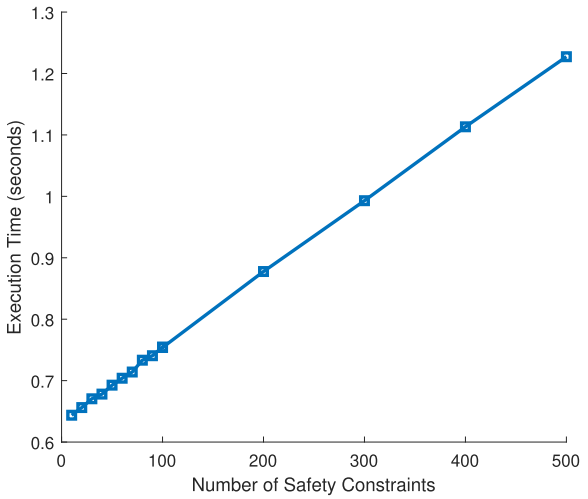

Fig. 9. Average execution time of the proposed scheme versus the number of safety constraints.

states metric computed according to our algorithm and reaches almost zero toward the end of the attack.

This demonstrates the usefulness of our algorithm in the worst case, where the actual state of the system significantly diverges from the real estimate under a stealthy attack. While this may be overly conservative when the system is not under attack, the safety criticality of the systems that we consider justifies the need to employ monitoring that can generate early warnings when a dangerous stealthy attack is taking place. Hence, the task of precomputing reachable sets under attack and using them for online monitoring is well justified.

Our monitoring approach can be used as part of an early warning system to improve situational awareness and potentially preserve important data relevant to an investigation that a stealthy attack should indeed cause damage. A full treatment of this aspect of our approach is, however, deferred for future work.

C. Performance and Scalability

We ran 100-h simulations of the TEP to test the performance of our approach. For performance testing purposes, we modified the algorithm so that all the states along the prediction horizon are visited. This represents the worst case scenario. We tested the ability of Algorithm 2 to scale with respect to: 1) the length of the prediction horizon (number of steps K) and 2) the number of safety constraints. We performed this testing on a machine with an Intel i7-9750H CPU clocked at 2.6 GHz and with 16 GB of RAM.

Fig. 8 shows the average time needed to perform state prediction and safety checking (see Algorithm 2) for each number of steps with a fixed number of safety constraints.[4] Results show that the performance of Algorithm 2 scales linearly with the number of steps required for state prediction. Furthermore, at $K = 1000$ steps, equivalent to approximately 30-min ahead-of-time prediction, the execution time of the safety checking algorithm is smaller than the sampling period (1.8 s).

In addition, we tested the ability of our algorithm to scale with the number of safety constraints. To this end, we fixed K at 500 steps (i.e., $\approx$15 min), and we ran 100-h simulations for each number of safety constraints. For the purposes of testing, we generated random half-spaces representing safety constraints. Results are shown in Fig. 9. The execution time scales linearly with the number of safety constraints and, with 500 safety constraints, is still less than the sampling time of

[4]For a 100-h simulation, this average is taken over approximately 2×10^5 safety checks for a sampling period of 1.8 s.

the system. Hence, the proposed algorithm exhibits excellent real-time performance in the presence of more complex safety constraints.

It is worth noting that the performance of the proposed algorithm can be improved significantly if implemented with a compiled language, such as C++, instead of MATLAB. This is the case in most control systems applications.

D. Implementation Challenges

Our evaluation has shown that the proposed monitoring approach exhibits high accuracy and real-time performance. When it comes to its implementation in practice, we note that the approach relies on a model of the system and its anomaly detector, which is normally available during the control design phase. The stealthy attack model that we focus on encompasses several optimized stealthy attack strategies proposed in previous work. This is also evident by the ability of Algorithm 2 to exhibit high accuracy with respect to a large number of randomized attacks in our validity tests (see Section VI-A). Finally, the attack model's relative practicality with respect to the attacker makes it a reliable heuristic threat model to consider for a practical design of monitoring algorithms.

In our approach, we require some design parameters to be set. For Algorithm 1, a choice for the size of the partitioning interval Δ_h must be made to guide the grid search. This parameter depends entirely on the desired tightness of the reachable set approximation. As this is a design-time activity, the computational cost of choosing an arbitrarily small Δ_h could be negligible when considering the increased tightness of the resulting approximation. Our experiments have shown that, even with a choice of $\Delta_h = 0.01$, we can achieve high accuracy in terms of safety checks. For Algorithm 2, the design choice is to set a proper length for the prediction horizon K. We showed that extensive randomized simulations performed in Section VI-A can help in choosing a value for K that achieves an acceptable tradeoff between how "early" we would like to raise warnings versus the accuracy of Algorithm 2 (see Figs. 2 and 3).

VII. Conclusion

In this article, we have presented a predictive online safety monitoring approach for LTI systems under potential stealthy sensor attacks. Our approach precomputes off-line symbolic reachable sets in terms of the system's state estimate by considering the evolution of the estimation error under a potential stealthy attack. Given the current state of the system and controllers, we predict, in real time, the control flow of the system for a certain number of steps in the future. The precomputed sets are then instantiated at the predicted estimates. We use ellipsoidal calculus techniques to perform emptiness checks of the intersection of the precomputed set with a set of unsafe states. We applied the approach to the large-scale TEP where we validated our approach and we showed that it can perform safety checks in a timely manner. Furthermore, we demonstrated the improvement over existing online monitoring techniques, and we showed that the computation of reachable sets under stealthy attacks is well justified in safety-critical applications. In the future, we will study in more detail the uncertainty propagation caused by the prediction of future states and its effect on the validity of the safety checking. In addition, we will study the possibility of extending the proposed online monitoring approach to other attack models, such as the replay or the covert attack, particularly in situations where it is justifiable to consider more resource-intensive stealthy attacks from the attacker's point of view. We also plan to consider systems that exhibit a high degree of nonlinearity and may not be modeled in the LTI framework.

References

[1] F. Akowuah and F. Kong, "Real-time adaptive sensor attack detection in autonomous cyber-physical systems," in *Proc. IEEE 27th Real-Time Embedded Technol. Appl. Symp. (RTAS)*, May 2021, pp. 237–250.

[2] D. Althoff, M. Althoff, and S. Scherer, "Online safety verification of trajectories for unmanned flight with offline computed robust invariant sets," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 3470–3477.

[3] M. Azzam, L. Pasquale, G. Provan, and B. Nuseibeh, "Grounds for suspicion: Physics-based early warnings for stealthy attacks on industrial control systems," 2021, *arXiv:2106.07980*.

[4] C.-Z. Bai, F. Pasqualetti, and V. Gupta, "Security in stochastic control systems: Fundamental limitations and performance bounds," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2015, pp. 195–200.

[5] A. Bathelt, N. L. Ricker, and M. Jelali, "Revision of the Tennessee Eastman process model," *IFAC-PapersOnLine*, vol. 48, no. 8, pp. 309–314, 2015.

[6] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[7] A. Carcano, A. Coletta, M. Guglielmi, M. Masera, I. N. Fovino, and A. Trombetta, "A multidimensional critical state analysis for detecting intrusions in SCADA systems," *IEEE Trans. Ind. Informat.*, vol. 7, no. 2, pp. 179–186, May 2011.

[8] J. H. Castellanos and J. Zhou, "A modular hybrid learning approach for black-box security testing of CPS," in *Proc. Int. Conf. Appl. Cryptogr. Netw. Secur.* Cham, Switzerland: Springer, 2019, pp. 196–216.

[9] X. Chen and S. Sankaranarayanan, "Model predictive real-time monitoring of linear systems," in *Proc. IEEE Real-Time Syst. Symp. (RTSS)*, Dec. 2017, pp. 297–306.

[10] Y. Chou, H. Yoon, and S. Sankaranarayanan, "Predictive runtime monitoring of vehicle models using Bayesian estimation and reachability analysis," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 2111–2118.

[11] A. Coletta, "Predictive detection of known security criticalities in cyber physical systems with unobservable variables," in *Proc. 11th Int. Conf. Secur. Appl. (CNSA)*, 2018, pp. 61–77.

[12] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Comput. Chem. Eng.*, vol. 17, no. 3, pp. 245–255, Mar. 1993.

[13] S. Etigowni, S. Hossain-McKenzie, M. Kazerooni, K. Davis, and S. Zonouz, "Crystal (ball) I look at physics and predict control flow! Just-ahead-of-time controller recovery," in *Proc. 34th Annu. Comput. Secur. Appl. Conf.*, 2018, pp. 553–565.

[14] Y. Geng, Y. Wang, W. Liu, Q. Wei, K. Liu, and H. Wu, "A survey of industrial control system testbeds," in *Proc. IOP Conf., Mater. Sci. Eng.*, vol. 569, 2019, Art. no. 042030.

[15] J. Giraldo *et al.*, "A survey of physics-based attack detection in cyber-physical systems," *ACM Comput. Surv.*, vol. 51, no. 4, p. 76, 2018.

[16] P. Griffioen, S. Weerakkody, B. Sinopoli, O. Ozel, and Y. Mo, "A tutorial on detecting security attacks on cyber-physical systems," in *Proc. 18th Eur. Control Conf. (ECC)*, Jun. 2019, pp. 979–984.

[17] Z. Guo, D. Shi, K. H. Johansson, and L. Shi, "Worst-case stealthy innovation-based linear attack on remote state estimation," *Automatica*, vol. 89, pp. 117–124, Mar. 2018.

[18] N. Hashemi, C. Murguia, and J. Ruths, "A comparison of stealthy sensor attacks on control systems," in *Proc. Annu. Amer. Control Conf. (ACC)*, Jun. 2018, pp. 973–979.

[19] N. Hashemi and J. Ruths, "Gain design via LMIs to minimize the impact of stealthy attacks," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2020, pp. 1274–1279.

[20] H. Hu, M. Fazlyab, M. Morari, and G. J. Pappas, "Reach-SDP: Reachability analysis of closed-loop systems with neural network controllers via semidefinite programming," 2020, *arXiv:2004.07876*.

[21] T. T. Johnson, S. Bak, M. Caccamo, and L. Sha, "Real-time reachability for verified simplex design," *ACM Trans. Embedded Comput. Syst.*, vol. 15, no. 2, p. 26, 2016.

[22] M. Krotofil and J. Larsen, "Rocking the pocket book: Hacking chemical plants," in *Proc. DefCon Conf., DEFCON*, 2015, pp. 1–52.

[23] A. B. Kurzhanski and P. Varaiya, "Ellipsoidal techniques for reachability analysis," in *Proc. Int. Workshop Hybrid Syst., Comput. Control*. Berlin, Germany: Springer, 2000, pp. 202–214.

[24] A. A. Kurzhanskiy and P. Varaiya, "Ellipsoidal toolbox (ET)," in *Proc. 45th IEEE Conf. Decis. Control*, 2006, pp. 1498–1503.

[25] C. Kwon and I. Hwang, "Reachability analysis for safety assurance of cyber-physical systems against cyber attacks," *IEEE Trans. Autom. Control*, vol. 63, no. 7, pp. 2272–2279, Jul. 2018.

[26] C. Kwon and I. Hwang, "Recursive reachable set computation for on-line safety assessment of the cyber-physical system against stealthy cyber attacks," in *Proc. 54th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2016, pp. 1123–1128.

[27] C. Kwon, W. Liu, and I. Hwang, "Security analysis for cyber-physical systems against stealthy deception attacks," in *Proc. Amer. Control Conf.*, Jun. 2013, pp. 3344–3349.

[28] T. Larsson, K. Hestetun, E. Hovland, and S. Skogestad, "Self-optimizing control of a large-scale plant: The Tennessee eastman process," *Ind. Eng. Chem. Res.*, vol. 40, no. 22, pp. 4889–4901, Oct. 2001.

[29] C. Le Guernic, "Reachability analysis of hybrid systems with linear continuous dynamics," Ph.D. dissertation, École Doctorale Mathématiques, Sci. et Technol. de l'Inf., Informatique, Univ. Grenoble I-Joseph Fourier, Grenoble, France, 2009.

[30] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the cyber attack on the Ukrainian power grid," Electr. Inf. Sharing Anal. Center (E-ISAC), 2016, pp. 1–29, vol. 388.

[31] J. Milosevic, D. Umsonst, H. Sandberg, and K. H. Johansson, "Quantifying the impact of cyber-attack strategies for control systems equipped with an anomaly detector," in *Proc. Eur. Control Conf. (ECC)*, Jun. 2018, pp. 331–337.

[32] J. Milošević, A. Teixeira, T. Tanaka, K. H. Johansson, and H. Sandberg, "Security measure allocation for industrial control systems: Exploiting systematic search techniques and submodularity," *Int. J. Robust Nonlinear Control*, vol. 30, no. 11, pp. 4278–4302, Jul. 2020.

[33] Y. Mo and B. Sinopoli, "On the performance degradation of cyber-physical systems under stealthy integrity attacks," *IEEE Trans. Autom. Control*, vol. 61, no. 9, pp. 2618–2624, Sep. 2016.

[34] S. Mokhatab and W. A. Poe, *Handbook of Natural Gas Transmission and Processing*. Houston, TX, USA: Gulf Professional, 2012.

[35] C. Murguia and J. Ruths, "On model-based detectors for linear time-invariant stochastic systems under sensor attacks," *IET Control Theory Appl.*, vol. 13, no. 8, pp. 1051–1061, 2019.

[36] C. Murguia and J. Ruths, "On reachable sets of hidden CPS sensor attacks," in *Proc. Annu. Amer. Control Conf. (ACC)*, Jun. 2018, pp. 178–184.

[37] C. Murguia, I. Shames, J. Ruths, and D. Nesic, "Security metrics of networked control systems under sensor attacks (extended preprint)," 2018, *arXiv:1809.01808*.

[38] F. Pasqualetti, F. Dorfler, and F. Bullo, "Control-theoretic methods for cyberphysical security: Geometric principles for optimal cross-layer resilient control systems," *IEEE Control Syst.*, vol. 35, no. 1, pp. 110–127, Feb. 2015.

[39] N. L. Ricker, "Decentralized control of the Tennessee Eastman challenge process," *J. Process Control*, vol. 6, no. 4, pp. 205–221, Aug. 1996.

[40] N. L. Ricker, "Model predictive control of a continuous, nonlinear, two-phase reactor," *J. Process Control*, vol. 3, no. 2, pp. 109–123, May 1993.

[41] R. S. Smith, "Covert misappropriation of networked control systems: Presenting a feedback structure," *IEEE Control Syst.*, vol. 35, no. 1, pp. 82–92, Feb. 2015.

[42] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," *Automatica*, vol. 51, pp. 135–148, Jan. 2015.

[43] H.-D. Tran, L. V. Nguyen, P. Musau, W. Xiang, and T. T. Johnson, "Decentralized real-time safety verification for distributed cyber-physical systems," in *Proc. Int. Conf. Formal Techn. Distrib. Objects, Compon., Syst.* Cham, Switzerland: Springer, 2019, pp. 261–277.

[44] D. Umsonst and H. Sandberg, "Anomaly detector metrics for sensor data attacks in control systems," in *Proc. Annu. Amer. Control Conf. (ACC)*, Jun. 2018, pp. 153–158.

[45] D. I. Urbina *et al.*, "Limiting the impact of stealthy attacks on industrial control systems," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 1092–1105.

[46] X.-G. Zhang, G.-H. Yang, and S. Wasly, "Man-in-the-middle attack against cyber-physical systems under random access protocol," *Inf. Sci.*, vol. 576, pp. 708–724, Oct. 2021.

[47] X.-G. Zhang and G. Yang, "Kullback–Leibler-divergence-based attacks against remote state estimation in cyber-physical systems," *IEEE Trans. Ind. Electron.*, vol. 69, no. 5, pp. 5353–5363, May 2022.