Worst-Case Impact Assessment of Multi-Alarm Stealth Attacks Against Control Systems with CUSUM-Based Anomaly Detection

Gabriele Gualandi
Mälardalen University, Sweden
Email: gabriele.gualandi@mdu.se

Alessandro V. Papadopoulos
Mälardalen University, Sweden
Email: alessandro.papadopoulos@mdu.se

***Abstract*—Manipulating sensor data can deceive cyber-physical systems (CPSs), leading to hazardous conditions in physical plants. An Anomaly Detection System (ADS) like CUSUM detects ongoing attacks by comparing sensor signals with those generated by a model. However, physics-based methods are threshold-based, which can result in both false positives and undetectable attacks. This can lead to undetected attacks impacting the system state and potentially causing large deviations from the desired behavior. In this paper, we introduce a metric called transparency that uniquely quantifies the effectiveness of an ADS in terms of its ability to prevent state deviation. While existing research focuses on designing optimal zero-alarm stealth attacks, we address the challenge of detecting more sophisticated multi-alarm attacks that generate alarms at a rate comparable to the system noise. Through our analysis, we identify the conditions that require the inclusion of multi-alarm scenarios in worst-case impact assessments. We also propose an optimization problem designed to identify multi-alarm attacks by relaxing the constraints of a zero-alarm attack problem. Our findings reveal that multi-alarm attacks can cause a more significant state deviation than zero-alarm attacks, emphasizing their critical importance in the security analysis of control systems.**

***Index Terms*—security, control systems, optimization**

I. Introduction

Autonomous systems rely heavily on control systems, which are investigated in this paper. Control theory techniques play a significant role in advancing autonomic computing, leading to more reliable and effective autonomous behavior. Self-defense is crucial in detecting and countering malicious attacks, ultimately increasing system integrity and functionality. This study aims to quantify the potential damage caused by successful attacks, with the goal of strengthening the resilience of autonomous systems. By understanding the maximum impact of hostile activities, we provide valuable insights that can guide the design process, ensuring the creation of systems capable of withstanding worst-case scenarios.

Control systems face increasing cyber threats, necessitating the development of intrusion detection methods for cyber-physical systems [1]. Detection processes involve measuring signal anomalies using a reference model, as redundant execution alone cannot guarantee detection if compromised components are not part of the control software. Physics-based detection is required for anomalies at interfaces, such as sensors and actuators [2]. In this context, Anomaly Detection Systems (**ADS**) compare interface signals between a physical system (Plant) and a model to identify anomalies. Pre-defined policies sanction the presence of an attack based on the run-time measure of anomaly. Detection processes are probabilistic due to unknown disturbances, model uncertainties, and numerical approximations (noise). A *residual* represents the difference vector between a signal from the system under control, called *Plant*, and one from the Plant's model. *Anomaly* is a measure of distance between a residual and the origin, which is subjected to a *threshold* to distinguish between H_0 and H_1 hypotheses (absence or presence of attacks).

This work advances the state-of-the-art in determining the *impact* of cyberattacks on control systems. The Worst Case Impact (**WCI**) quantifies the effects of an attack in terms of state deviation. As defined in [3], [4], the WCI represents the maximum displacement of a Plant's state from the setpoint induced by an attack. As discussed in [5], WCI quantifies a state deviation independent of a specific attack strategy (e.g., denial of service, rerouting, sign alternation or replay).

As part of a WCI assessment, the *worst-case* scenario considers powerful attackers with knowledge of the parameters of the control system and the capability to compromise a significant number of devices (e.g., all sensors). If the attacker knows the specifics of the ADS (s)he can remain undetected to maximize the impact (*stealth attacks*). Therefore, the presence of an ADS could limit the impact of an attack, provided that the attacker intends to remain stealthy. It is, therefore, critical to perform a WCI assessment before deploying a control system in the real world (especially for critical control systems) since it can sanction the effectiveness of the ADS in limiting the effects of stealth attacks.

CUSUM, a change-point detection procedure, has historically been employed for the monitoring of industrial processes. In recent years, its application has expanded to encompass anomaly detection aimed at enhancing security across a diverse range of systems. Examples of these novel applications span various domains, including autonomous systems [6], controller area networks [7], and control systems [8]. In a CUSUM ADS, anomalies are accumulated over time, triggering an *alarm*

This work was supported by the Swedish Research Council (VR) with the PSI project (No. #2020-05094), by the Knowledge Foundation (KKS) with the FIESTA project (No. #20190034).

979-8-3503-3744-0/23/$31.00 ©2023 IEEE
DOI 10.1109/ACSOS58161.2023.00029

when the threshold $\tau \in \mathbb{R}^{\geq 0}$ is surpassed. The Average Run Length under H_0 hypotheses, ARL^0, is the ratio between the length of an observation window and the number of alarms, under the assumption that no attack is underway. In other words, ARL^0 measures the average number of time samples between successive false positives, which are unavoidable due to noise. Increasing the threshold τ can result in a longer ARL, but it reduces sensitivity to anomalies. Therefore tuning the threshold τ involves a trade-off between shorter ARL and sensitivity to anomalies.

A. Related works

Generally, the assessment of WCI entails solving optimization problems that simultaneously model the control system and the ADS, with the objective of maximizing the deviation in the Plant's state caused by the compromised control signal(s). However, the methods for identifying solutions to the WCI are restricted to a limited number of detection systems that can be encapsulated as constraints within an optimization problem, such as the chi-squared [9] and CUSUM [8] ADSs.

A CUSUM-based ADS is modeled in [8] to identify stealth attacks that maximize the infinity norm of the Plant's state deviation. By reformulating a problem as convex, the authors determine the globally optimal solution (i.e., the most dangerous attack). In [10], the authors introduce a novel metric named the *impact of undetected attacks*. They argue that metrics such as the false-positive rate may be misleading since an omniscient attacker can always remain undetected. They consider a stealth attack's maximum deviation *per second* and present a new paradigm for tuning a CUSUM detection system. Instead of tuning the detection threshold as a trade-off between the false positive rate and attack sensitivity, they propose considering ARL as a trade-off between the usability of a detection system and deviation per second due to stealth attacks. While a short ARL can limit the impact of stealth attacks by generating many false positives, it may also reduce the usability of the ADS.

In [11], [12], the authors perform a WCI assessment for a control system using a steady-state Kalman filter and a CUSUM detection system. In their simulated experiments, they decrease the ARL to the limit where the ADS is "practically usable", i.e., they use different ARL values spanning from 0.25 to 0.02 seconds (which we consider a short ARL). However, they do not quantify the advantages of decreasing the ARL.

In [13], the authors introduce a distinction between different kinds of stealth attacks: *zero-alarm* attacks aim never to trigger an alarm, while *multi-alarms* attacks, sometimes referred as to hidden attack, produce alarms at a rate similar to the nominal one, thus mimicking the nominal ARL. According to the study, there is a need to go beyond traditional detectors to detect these threats.

B. Contributions

To the best of our knowledge, no methods are currently available in the literature for determining optimal multi-alarm attacks considering an entire attack duration. The works mentioned in Section I-A provide solutions for zero-alarm attacks. Furthermore, the literature needs to address why it might be necessary to consider multi-alarm attacks instead of zero-alarm. This work has the following research questions:

- **RQ1**: in which cases should multi-alarm attacks be considered (instead of zero-alarm) in a Worst Case Impact assessment?
- **RQ2**: how to find multi-alarm attacks?

For RQ1, we introduce a metric called *transparency*, which measures how well an ADS limits stealth attacks; the greater the transparency, the lower the ability of an ADS to prevent state deviation. We then consider tuning a CUSUM-based ADS, specifically choosing a threshold τ. We demonstrate that to reduce the transparency of an ADS protecting a mass-spring-damper system to acceptable levels, the value of τ may need to decrease, resulting in a short ARL (e.g., 30 samples at a sampling rate of 0.05 seconds). When the ARL is very short, the WCI should not be measured by considering only zero-alarm attacks, as this might underestimate the true impact of attacks that are allowed to trigger alarms occasionally. In other words, attacks that can trigger alarms occasionally (i.e., can produce greater anomaly) have the potential to deviate the state more than attacks that are constrained by design not to generate any alarm.

For RQ2, we propose to modify a problem designed for zero-alarm attacks to allow occasional alarms without significantly deviating from the nominal ARL.

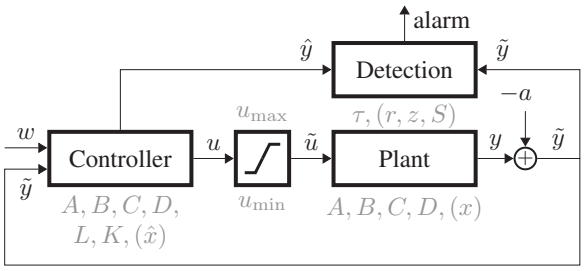

Fig. 1: Control system, under sensor attack a, protected by an attack detection system.

II. Model and assumptions

A. Control system and ADS

Consider a discrete-time feedback control system composed of a physical Plant and a controller, subject to a "sensor attack" [14], where the attacker can read and write to all sensor readings. Fig. 1 illustrates the closed-loop and detection systems. In gray, we highlight specific block parameters and, in parenthesis, internal blocks state variables. The system is sampled at prescribed times, indicated with $k \in \mathbb{Z}$. We denote with x_k the (internal) state of the Plant at time k, and with x_k^j, the value of its j-th component. At every sampling instant k, the controller receives a measurement $\tilde{y}_k$ of the Plant output y_k and produces an actuation signal u_k, to drive the future Plant's

state towards a reference value, w_k passed to the controller as input (setpoint). The value of the control signal u_k is saturated to belong to the interval $[u_{\min}, u_{\max}]$ due to physical limitations, generating $\tilde{u}_k$. We assume that an attacker can tamper with the system behavior by forging an attack signal a_k that is added to the actual measurement of the output,

$$\tilde{y}_k = y_k + a_k. \tag{1}$$

According to our threat model, the goal of the attacker is to to divert the state of the Plant from its desired value w_k to reach a dangerous state.

We assume the Plant is linear, time-invariant, controllable, and observable, with discrete-time dynamic equations

$$\text{Plant} = \begin{cases} x_{k+1} &= A\,x_k + B\,\tilde{u}_k + \upsilon_k \\ y_k &= C\,x_k + D\,\tilde{u}_k + \psi_k \end{cases} \tag{2}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, $D \in \mathbb{R}^{p\times m}$, n is the number of states, m is the number of inputs, p is the number of outputs, and i.i.d. multivariate zero-mean Gaussian noises $\upsilon \in \mathbb{R}^n$ and $\psi \in \mathbb{R}^p$ with covariance matrices respectively $\Upsilon \in \mathbb{R}^{n\times n}$ and $\Psi \in \mathbb{R}^{p\times p}$.

We consider a state-feedback controller with a steady state Kalman filter as in [12], but, in addition, we also consider the actuator's saturation. Specifically, assuming without the loss of generality setpoint $w_k = 0$ the controller equations are

$$\text{Controller} = \begin{cases} r_k &= \tilde{y}_k - \hat{y}_k \\ \hat{x}_{k+1} &= A\,\hat{x}_k + B\,\tilde{u}_k + L\,r_k \\ \hat{y}_k &= C\,\hat{x}_k + D\,\tilde{u}_k \\ u_k &= -K\hat{x}_k \\ \tilde{u}_k &= \text{sat}\,(u_k) \end{cases} \tag{3}$$

where $\hat{x}_k$ and $\hat{y}_k$ are respectively the estimated Plant state and output, and u_k and $\tilde{u}_k$ are the control signals respectively produced by the controller and received by the Plant, due to saturation levels. The unique stabilizing solution P (covariance matrix), state-feedback gain K, and closed-loop eigenvalues are obtained by solving the following algebraic Riccati equation:

$$APA^T - P - (APC^T)(\Psi + CPC^T)^{-1}(APC^T)^T + \Upsilon = 0$$

i.e., $L = (APC^T)(\Sigma + CPC^T)^{-1}$ where $\Sigma = E[r_{k+1}] = CPC^T + \Psi$, $\Sigma \in \mathbb{R}^{p\times p}$, is the covariance of signal r. We assume Σ positive-definite (a standard assumption that guarantees the convergence of the Kalman filter employing P). The function $\text{sat}(\cdot)$, saturates the control signal within the values $[u_{\min}, u_{\max}]$. We assume the saturation constraints are symmetric ($u_{\max} = -u_{\min}$).

We consider a non-parametric CUSUM ADS [15] with equations

$$\text{Detection} = \begin{cases} z_k &= r_k^T \Sigma^{-1} r_k \\ E_k &= \max\,(0, S_k + z_k - b) \\ S_{k+1} &= \begin{cases} E_k & \text{if } E_k \le \tau \\ 0 & \text{if } E_k > \tau \end{cases} \\ \mathbf{C}_k &= \begin{cases} 0 & \text{if } E_k \le \tau \\ 1 & \text{if } E_k > \tau \end{cases} \\ S_0 &= 0 \end{cases} \tag{4}$$

where the distance measure z_k is the squared residual r_k scaled by the inverse of its covariance matrix Σ [11], [12], an expression also known as the squared Mahalanobis distance. The output of the CUSUM is a binary sequence $\mathbf{C}$ flagging alarm times. When the sum S exceeds the threshold τ, the detector generates an alarm and resets S to a value of zero.

Our work considers sequences of numbers as sets with order preserved by operation. We denote with $\mathbf{S}_1 \setminus \mathbf{S}_2$ the difference between sets $\mathbf{S}_1$ and $\mathbf{S}_2$. For a set $\mathbf{S}$, we define $|\mathbf{S}| = \sum_{s\in\mathbf{S}} 1$ (number of elements) and $|\mathbf{S}|^{=1} = \sum_{s\in\mathbf{S}:s=1} 1$ (number of elements equal to 1).

The ARL^0 is the Average Run Length of the CUSUM under the H_0 hypothesis (absence of attacks). Having a sufficiently long alarm sequence $\mathbf{C}$, then

$$\text{ARL}^0 = \frac{|\mathbf{C}| - |\mathbf{C}|^{=1}}{|\mathbf{C}|^{=1}} \tag{5}$$

where $|\mathbf{C}| - |\mathbf{C}|^{=1}$ is the number of times τ is not exceeded. Before deployment, the ARL^0 has to be tuned by selecting the CUSUM parameters τ and b. The Σ^{-1} factor in the definition of z_k rescales the distribution so that the CUSUM parameters can be assigned independent of the specific statistics of the noises υ_k, ψ_k. More specifically, being in our case r_k a Gaussian distribution with zero mean, z_k follows a chi-squared distribution with p degrees of freedom. Therefore the expected value of z_k is equal to the dimensionality of r_k (see [11], [12]) e.g., if r has one dimension, then $\mathbf{E}[z_k] = 1$. To ensure mean square boundedness of S independently on τ, the CUSUM parameter b must be selected to be larger than $\mathbf{E}[z]$. On the other hand, as the value of b increases, the capacity of minor deviations to impact S diminishes, thereby reducing the sensitivity of the detection process. As a consequence, a good choice for b is a slight excess over the dimensionality of r, e.g., if r has one dimension, then $b = 1.01$ (as done in [12]).

Obtaining many run-length samples through simulation is generally difficult because run times can be extremely long. Knowing the expected value of z_k enables estimating through a Markov Chain the ARL resulting from a given τ (see [12]) - we call function tau2ARL(τ, b). Subsequently, it is possible to obtain the CUSUM τ as a function of a (desired) ARL^0 as the solution of

$$\underset{\tau}{\text{minimize}} \left(\text{ARL}^0 - \text{tau2ARL}(\tau, b)\right)^2. \tag{6}$$

B. Attacker

We are analyzing an attack with duration N steps. The attack steps are indexed by $i \in (1, \ldots, N)$. The attacker aims to maximize the impact at the final step N. The attacker has access to system matrices (A, B, C, D), controller parameters (L, K, sat), and ADS parameters (τ, b), and can read and write sensor data to remain undetected. We assume that the Plant's state is near the setpoint at the start of the attack, and the attacker plans an open-loop attack sequence without considering random noise. While a closed-loop strategy could be employed to counter or take advantage of noise, we assume the noise is not significant, and the planned attack's impact

cannot be considerably greater than the actual one. We also assume that the attacker plans the attack based on the Plant's state at the setpoint, but the actual state may differ slightly at the start of the attack. This paper does not discuss specific methods for realizing closed-loop attacks.

III. Optimization problems

A WCI assessment aims to determine whether an attack might lead to at least one dangerous state. The *maximal impact* is the solution to the problem

$$\text{maximize}\, \mathcal{D} \tag{7}$$

where $\mathcal{D}$ is a measure of impact. The optimization variables and constraints in Eq. (7) depend on the considered impact and the model of the control system. We adopt the same approach used by [5], [8], where they consider deviation the infinity norm of the Plant's state. In particular, [8] introduces a normalization matrix T_{norm}, i.e., $\mathcal{D} = ||T_{\text{norm}}\, x_N||_\infty$. A control system is considered safe if the maximal impact cannot displace any of the components of $T_{\text{norm}} x_N$ over a value of 1, which, according to [8], is equivalent to determining the maximum deviation for the absolute value of each component of x_N. To find the most dangerous attack lasting N, we create a separate optimization problem for each component of x throughout the entire duration. This means we solve n optimization problems having objective functions $\mathcal{D}_j = |x_N^j|, j = 1, \cdots, n$.

Let us define the sets of variables $\mathcal{X}$ and parameters $\mathcal{P}$

$$\mathcal{X} = \{a_i, x_i, \hat{x}_i, u_i, \tilde{u}_i, y_i, \hat{y}_i, \tilde{y}_i, r_i, z_i, S_i \mid i \in \{1, \cdots, N\}\}$$
$$\mathcal{P} = \{N, A, B, C, D, K, L, \Sigma^{-1}, u_{\max}, u_{\min}, \tau, b, \text{sat}, \Omega\}$$

where Ω is a set of integers in the range $(1, \cdots, N-1)$ (possibly empty, i.e., $|\Omega| = 0$). We define the following optimization problem relative to the j-th component of the Plant's state variable x (***Problem 1***):

$$\underset{\mathcal{X}}{\text{maximize}} \quad \mathcal{D}_j \tag{8a}$$
subject to:
$$x_1 = 0 \tag{8b}$$
$$\hat{x}_1 = 0 \tag{8c}$$
$$x_{i+1} = A\, x_i + B\, \tilde{u}_i \qquad i = 1, \cdots, N-1 \tag{8d}$$
$$y_i = C\, x_i + D\, \tilde{u}_i \qquad i = 1, \cdots, N \tag{8e}$$
$$\tilde{y}_i = y_i + a_i \qquad i = 1, \cdots, N \tag{8f}$$
$$\hat{x}_{i+1} = A\, \hat{x}_i + B\, \tilde{u}_i + L\, r_i \quad i = 1, \cdots, N-1 \tag{8g}$$
$$\hat{y}_i = C\, \hat{x}_i + D\, \tilde{u}_i \qquad i = 1, \cdots, N \tag{8h}$$
$$u_i = -K\, \hat{x}_i \qquad i = 1, \cdots, N \tag{8i}$$
$$r_i = \tilde{y}_i - \hat{y}_i \qquad i = 1, \cdots, N \tag{8j}$$
$$z_i = r_i^T \Sigma^{-1} r_i \qquad i = 1, \cdots, N \tag{8k}$$
$$\tilde{u}_i = \text{sat}(u_i) \qquad i = 1, \cdots, N \tag{8l}$$
$$0 \le S_i \le \tau \qquad i = 1, \cdots, N \tag{8m}$$
$$S_{i+1} \ge S_i + z_i - b \qquad i \in \{1, \ldots, N-1\} \setminus \Omega \tag{8n}$$
$$S_1 = 0 \tag{8o}$$
$$S_{i+1} \ge z_i - b \qquad i \in \Omega \tag{8p}$$

An *instance* $\mathcal{I}$ of Problem 1 provides a solution

$$\mathcal{X}^* = \{a_i^*, x_i^*, \hat{x}_i^*, u_i^*, \tilde{u}_i^*, y_i^*, \hat{y}_i^*, \tilde{y}_i^*, r_i^*, z_i^*, S_i^*\}.$$

based on parameters $\mathcal{P}$.

Eqs. (8b) and (8c) model the state of the controller when the attack begins according to the assumptions of Section II-B, Eqs. (8d) to (8f) are equations of the Plant under attack, and Eqs. (8g) to (8k) are the controller equations.

Eqs. (8m) to (8p) model the CUSUM dynamics. The set of indexes in Eq. (8n) is complementary to the set of indexes in Eq. (8p). The purpose of having $|\Omega| > 0$, so that Eq. (8p) is used, is relative to the multi-alarm case and is clarified in Section V. If $|\Omega| = 0$, so that Eq. (8p) is not used, it is determined a convex reformulation of Eq. (4) for zero-alarm attacks, as proved in [8]. The reformulation is valid because, assuming $S_1 = S_1^*$ and $z_i = z_i^*, \forall i$, then we have

$$S_i = S_i^*, \forall i \in (1, \cdots, N)$$

where S_i describes the actual CUSUM ADS as defined in Eq. (4), while S_i^* comes from the solution of $\mathcal{I}$. Note that assuming $x_1 = x_1^*$ and $\hat{x}_1 = \hat{x}_1^*$, the assumption $z_i = z_i^*$ necessarily holds because the constraints of $\mathcal{I}$ models exactly all the relations between variables in eq. (1)–(4) . The advantage of using constraints Eqs. (8m) to (8o) instead of Eq. (4) is that the former provide linear constraints instead of non-smooth constraints that the *max* operator would determine.

An instance $\mathcal{I}$ with $|\Omega| = 0$ is similar to the one defined by [8], where they obtain a convex problem to find the optimal zero-alarm attack. However, in our problem: i) we employ Σ^{-1} in the residual distance, as in [12], and ii) we also consider the actuator's saturation. The residual distance of Eq. (8k) does not affect convexity because the covariance matrix Σ is non-singular and positive semi-definite. Eq. (8l) describes the actuators' saturation. The resulting attack can be more or less effective depending on how $\text{sat}(\cdot)$ is specialized. There are at least two different ways to model the saturation [16]:

Overflow-prevent (Opt-P) constraint:

$$\begin{aligned} &u_{\min} \le \tilde{u}_i \le u_{\max}, \qquad i = 1, \cdots, N \\ &\tilde{u}_i = u_i, \qquad\qquad\quad i = 1, \cdots, N \end{aligned} \tag{9}$$

Overflow-allow (Opt-A) constraint:

$$\tilde{u}_i = \max\left(\min\left(u_i, u_{\max}\right), u_{\min}\right), \qquad i = 1, \cdots, N \tag{10}$$

Using Opt-A results in a more relaxed optimization problem compared to Opt-P, implying a potential to generate more dangerous attacks. However, a significant drawback of Opt-A is that it leads to a non-convex optimization problem, specifically a mixed-integer linear programming problem, thus making it challenging to identify a globally optimal solution. Conversely, Opt-P, being linear, does not impede convexity.

To summarize, if an instance $\mathcal{I}$ has $|\Omega| = 0$, then an optimal zero-alarm attack sequence $(a_1^*, \cdots, a_N^*)$ can be generated. Using Opt-P will result in a convex problem, while using Opt-A may lead to a better solution, but it is non-convex.

One method to perform a WCI assessment is to find a baseline solution using Opt-P, then use Opt-A to improve the baseline solution progressively.

IV. Transparency of an ADS

An ideal anomaly detection system (ADS) should have a low false positive rate (a long ARL^0) and be highly sensitive to attacks. However, increasing the τ to achieve a long ARL^0 can decrease sensitivity. There is a need for a unified definition of sensitivity in the literature. One informal interpretation of sensitivity is the ability to detect attacks amidst background noise. However, since the specific attack is unknown beforehand, a sensitivity value must be specified for each possible attack. To quantify the impact of stealth attacks and the effectiveness of an ADS, we introduce a new metric called *transparency*. We define transparency as the percentage of maximal impact an attacker can achieve despite the presence of the ADS. A highly transparent ADS becomes irrelevant to the attacker regarding the impact, and therefore such ADS is ineffective.

Having an instance $\mathcal{I}$ with solution $\mathcal{X}^*$, we denote with $\mathcal{X}^{\mathrm{U}}$ the solution of a modified instance of $\mathcal{I}$ where all constraints modeling the ADS are removed, that is, constraints (8m)–(8p). We define the transparency of an ADS (relative to a certain Plant's state component) as

$$\mathcal{T} = \frac{|x^*_{N-1}|}{|x^U_{N-1}|} \cdot 100 \in (0, 100]. \tag{11}$$

Maximum impact uniquely defines transparency, and the value of transparency remains invariant if deviation per unit of time is used (instead of absolute deviation), as proposed by [10].

Using simulated experiments, we demonstrate how transparency decreases with ARL^0 in zero-alarm attacks ($|\Omega| = 0$). As a Plant, we consider a simulated mass-spring-damper system having, in continuous time, equations

$$\begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = -\frac{k}{m} x_1(t) - \frac{d}{m} x_2(t) + \frac{1}{m} u(t) \\ y(t) = x_1(t), \end{cases} \tag{12}$$

with mass m kg, elastic constant k N/m and damping d Ns/m. The Plant has $m = 0.4$, $k = 0.15$, $d = 0.095$. Noises covariance matrices are $\Upsilon = [8.3,\ 0;\ 0,\ 8.3] \cdot 10^{-3}$, $\Psi = 0.08$. The model is discretized through the zero-order hold with sampling period $T_s = 0.05$ s, resulting in $\Sigma = 0.11$. In nominal conditions (the Plant is on the setpoint), the covariance matrix of the Plant's state x is $[0.41, -0.081; -0.081; 0.39]$.

To ensure a fair comparison across multiple solutions, in what follows, we consider only the constraints of Opt-P for the actuator's saturation as they result in convex problems. Thus, we are comparing the globally optimal solutions of each problem. We noticed that employing Opt-A produces similar results to those of Opt-P, affirming the applicability of our considerations. We observed that depending on the specific set of parameters considered, the use of Opt-A can lead to a very long computation time and possibly an increase in the introduced deviation up to approximately 15%.

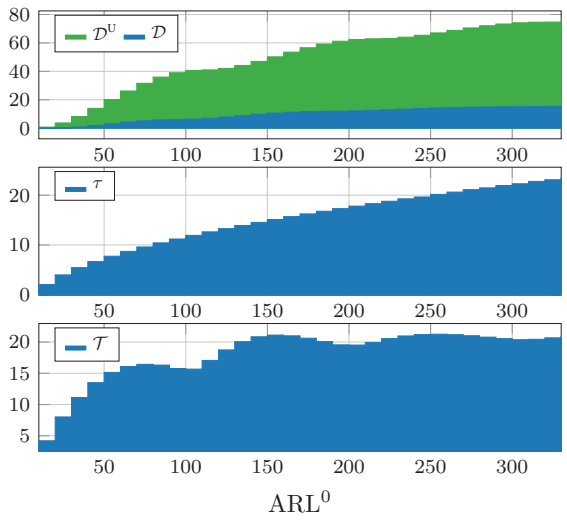

Fig. 2: Effects of changing ARL^0 in a zero-alarm attack scenario, with attack duration N equal to ARL^0. Smaller ARL^0 reduces the impact ($\mathcal{D}$), the threshold τ and the transparency ($\mathcal{T}$). In all the simulations, the actuator's saturation $u_{\max} = -u_{\min} = 4$.

We configure an instance $\mathcal{I}$ as in Problem 1, with $|\Omega| = 0$ (zero-alarm), and focusing on maximizing the value of the first component of the state, x_1 (which corresponds to the position of the mass). In the experiments, we tune τ to deliver values of ARL^0 equal to N and find solutions $\mathcal{X}^{\mathrm{U}}$ (absence of ADS) and $\mathcal{X}^*$ (presence of ADS). Let $\mathcal{D}^{\mathrm{U}}$ and $\mathcal{D}$ denote the respective impacts of $\mathcal{X}^{\mathrm{U}}$ and $\mathcal{X}^*$.

As N varies, the results are presented in Fig. 2. If ARL^0 exceeds 150 samples (7.5 seconds), the transparency reaches above 20%, implying an impact reduction of 80% due to the presence of the ADS. To attain a transparency of 10%, which is preferable since it denotes a 90% reduction in impact, it becomes necessary that ARL^0 falls below 60 samples.

V. Multi-alarm attack scenario

Ideally, CUSUM can be tuned to have a very long ARL^0 so that hypothesis H_0 can be set False upon any alarm. However, a long ARL^0 might make the transparency of the ADS unacceptable. We denote a *multi-alarm* scenario where the ARL^0 must be short (e.g., 30 samples) to obtain acceptable transparency. In a multi-alarm scenario, false positives are expected; therefore, it is necessary to monitor the alarm sequence $\mathbf{C}$ using an additional detector of higher level [17] monitoring $\mathbf{C}$ in a sliding window of length $M \in \mathbb{N}^+$ to decide if H_0 is False. We consider a high-level detector that monitors the difference between the observed ARL and the nominal ARL^0, defined as

$$e = \mathrm{ARL}^0 - \frac{M - |\mathbf{C}|^{=1}}{|\mathbf{C}|^{=1}} \in \mathbb{R} \tag{13}$$

Note that $e = 0 \implies |\mathbf{C}|^{=1} = \frac{M}{\mathrm{ARL}^0 + 1}$, which could be impossible for round-off error because $|\mathbf{C}|^{=1} \in \mathbb{N}^{\geq 0}$. Therefore, we define the closest integer approximation of the average number of alarms under H_0 as

$$|\mathbf{C}|^{=1}_{\text{TARGET}} = \text{round}\left(\frac{M}{\mathrm{ARL}^0 + 1}\right). \tag{14}$$

There is the following policy for the higher lever detector. $\mathrm{H}_0 =$ True if and only if the two following conditions are both True

$$|\mathbf{C}|^{=1} \leq |\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^{\mathrm{D}} \tag{15}$$

and

$$|\mathbf{C}|^{=1} \geq |\mathbf{C}|^{=1}_{\text{TARGET}} - \delta^{\mathrm{D}} \tag{16}$$

where $\delta^{\mathrm{D}} \in \mathbb{N}^{\geq 0}$ is a tolerance threshold on the number of alarms in the observation window. An attack is characterized as *hidden* if the correspondent alarm sequence $\mathbf{C}$ satisfies the condition $\mathrm{H}_0 =$ True.

Let us define a function $\mathcal{E}$ that describes the *effect* of an attack (without considering noise), implemented using recursive equations (1)–(4) with $\upsilon_k = 0$, $\psi_k = 0$ and considering $k = 1$ as the onset of the attack.

$$\mathcal{E}\left(x_1, \hat{x}_1, S_1, (a_1, \cdots, a_N)\right) \mapsto \mathcal{X} \cup \mathbf{C}. \tag{17}$$

where $(a_1, \cdots, a_N)$ is an attack sequence. Function $\mathcal{E}$ yields all the values of the variables in $\mathcal{X}$ and $\mathbf{C}$ obtained from the recursive equations (actual control system and ADS), assuming there is no noise.

There are the following notational conventions to enhance brevity in the presentation. A superscript $*$ on a variable indicates its association with the solution $\mathcal{X}^*$ of an instance $\mathcal{I}$ of Problem 1. When a sampling time subscript is absent, this signifies a sequence that persists throughout the entire duration of the attack. For instance, a^* refers to an optimal attack sequence $(a_1^*, \cdots, a_N^*)$. Furthermore, whenever variables from $\mathcal{X}$ or $\mathbf{C}$ are used without any supplementary notation, it is implicitly understood that they are derived by supplying $\mathcal{E}$ with the attack sequence a^* extracted from a solution $\mathcal{X}^*$. For example, S embodies the values of the actual CUSUM sum as computed by $\mathcal{E}$ upon providing a^*. More precisely, the effects of a solution $\mathcal{X}^*$ are

$$\mathcal{X} \cup \mathbf{C} = \mathcal{E}\left(x_1, \hat{x}_1, S_1, a^*\right). \tag{18}$$

Lemma V.1. *Consider a solution $\mathcal{X}^*$, and its effect $\mathcal{X} \cup \mathbf{C}$. Assuming $x_1^* = x_1$ and $\hat{x}_1^* = \hat{x}_1$, we have:*

$$\mathcal{X}^* \setminus S^* = \mathcal{X} \setminus S$$

independently on the content of Ω.

Proof. Constraints of $\mathcal{I}$ models exactly all the variables of eq. (1)–(4) except for S (independently on the content of Ω). Consequently, Eq. (18) yields $\mathcal{X}$, where, relative to $\mathcal{X}^*$, discrepancies can only be introduced between S_k^* and S_k for some k. Simultaneously, all other corresponding variables in $\mathcal{X}^*$ and $\mathcal{X}$ must retain identical values for each time k. □

According to the assumptions of Section II-B, we can affirm that $x_1 = x_1^* = 0$ and $\hat{x}_1 = \hat{x}_1^* = 0$. Therefore Lemma V.1 assures that Eq. (18) correctly represents the Plant's dynamics, regardless of Ω. In the presence of process noise, the actual evolution of the Plant's state may differ from x^*, but we assume the influence is negligible. On the other hand, sensor noise does not introduce any changes in the planned attack because the attacker can directly inject $\tilde{y}^*$, independently on y.

A. Upper bound on the number of alarms

In this section, we modify the optimization problem for zero-alarm attacks to provide a solution guaranteeing that condition of Eq. (15) is not violated.

Lemma V.2. *Consider an instance $\mathcal{I}$ and sequence (interval) of time instants $\Delta = (\Delta_s, \cdots, \Delta_e)$ with at least two elements, chosen such that*

$$\forall k \in (\Delta_s, \cdots, \Delta_e),\ k \notin \Omega.$$

Assuming $S^*_{\Delta_s} \geq S_{\Delta_s}$ we have:

$$S_k^* \geq S_k,\ k \in (\Delta_s + 1, \cdots, \Delta_e).$$

Proof. In simpler terms, when instance $\mathcal{I}$ uses Eq. (8n) within an interval, and the initial value of S^* does not fall below that of the initial CUSUM sum S, then S^* will persistently dominate S throughout the entire interval.

Within this proof, we consider without the loss of generality $\Delta_s = 1$. Hence we rewrite the assumption as $S_1^* \geq S_1$.

For Eq. (4), either $S_2 = 0$ or $S_2 > 0$. If $S_2 = 0$, for Eq. (8m) $S_2^* \geq 0 = S_2$ and the proof is valid in $k = \Delta_s + 1 = 2$.

If, instead, $S_2 > 0$, for Eq. (4) we have $S_2 = S_1 + v_1$, where we introduced $v_1 = z_1 - b \in \mathbb{R}$. Define also the discrete difference in the solution $v_1^* = z_1^* - b$. By virtue of Lemma V.1, it follows that $v_1^* = v_1$, and Eq. (8n) becomes $S_2^* \geq S_1^* + v$. Defining $\delta_1 = S_1^* - S_1$, we have $S_2^* \geq \delta_1 + S_1 + v_1$. Using $S_2 = S_1 + v_1$ we have $S_2^* \geq \delta_1 + S_2$. Being $\delta_1 \geq 0$ for the hypothesis, then $S_2^* \geq S_2$, hence the proof is valid in $k = \Delta_s + 1 = 2$.

By induction, the result holds for the entire sequence (i.e., by considering $\delta_2 = S_2^* - S_2 \geq 0$ and so on). □

Lemma V.3. *Consider an interval Δ as defined in Lemma V.2. Assuming $S^*_{\Delta_s} \geq S_{\Delta_s}$, no alarm can be generated during Δ, i.e., $(\mathbf{C}_{\Delta_s}, \cdots, \mathbf{C}_{\Delta_e}) = (0, \cdots, 0)$.*

Proof. Lemma V.2 and Eq. (8m) guarantees that $\tau \geq S^* \geq S$ during Δ. Therefore, no alarms can be generated during Δ, as S cannot exceed the threshold τ. □

For Lemma V.3, the solution of an instance $\mathcal{I}$ wherein $|\Omega| = 0$, and assuming $S^*_{\Delta_s} \geq S_{\Delta_s}$, will result in a zero-alarm attack. This conclusion is drawn because we can apply Lemma V.3 over the entire attack duration. Conversely, an instance $\mathcal{I}$ where $|\Omega| > 0$ does not establish a unique interval Δ in which only Eq. (8n) is used. Therefore, we cannot apply Lemma V.3 across the full attack duration. The condition $|\Omega| > 0$ may prompt S to go beyond τ, generating alarm(s),

even if S^* does not exceed τ. In what follows, we determine the maximum number of alarms and discuss their location within the alarm sequence $\mathbf{C}$ depending on $|\Omega|$.

Lemma V.4. *During an interval Δ defined as in Lemma V.2 there can be generated a maximum of one alarm by the solution of $\mathcal{I}$, independently on the value of $S^*_{\Delta_s}$ and S_{Δ_s}.*

Proof. Suppose that S generates at least one alarm during Δ, and denote the instant of the first alarm as $l \geq 1$. For Eq. (4) the CUSUM resets at $k = l + 1$, then can affirm that $S_{l+1} = 0 \leq S^*_{l+1}$. According to Lemma V.3, no further alarms can occur during the interval $(l + 1, \cdots, \Delta_e)$. □

Lemma V.5. *Consider an arbitrary interval $\tilde{\Delta}$ chosen such that*

$$\tilde{\Delta} = (\Delta_s - 1, \Delta_s, \cdots, \Delta_e) \tag{19}$$

where $\Delta_s - 1 \in \Omega$ and $\Delta = (\Delta_s, \cdots, \Delta_e) \notin \Omega$. Then, during the interval $\tilde{\Delta}$, the solution of $\mathcal{I}$ can generate a maximum of one alarm.

Proof. Interval $\tilde{\Delta}$ is defined such that the instance $\mathcal{I}$ utilizes Eq. (8p) at the initial instant and Eq. (8n) for the remaining instants. In the case where an alarm is generated at $k = \Delta_s - 1$, the CUSUM resets and we can affirm that $S_{\Delta_s} = 0 \leq S^*_{\Delta_s}$. Based on Lemma V.3, no further alarms can be triggered in the interval $\Delta = (\Delta_s, \cdots, \Delta_e)$. Alternatively, if no alarm is generated at $k = \Delta_s - 1$, for Lemma V.4 a maximum of one alarm can be generated in $\Delta = (\Delta_s, \cdots, \Delta_e)$. □

Theorem V.6. *The number of alarms generated by the solution of an instance $\mathcal{I}$ is less or equal to the number of instants in which $\mathcal{I}$ uses Eq.* (8p) *plus one unit, i.e.,*

$$|\mathbf{C}|^{=1} \leq |\Omega| + 1$$

Proof. If $|\Omega| = 0$, then $|\mathbf{C}|^{=1} \leq 1$ for Lemma V.4. If $|\Omega| > 0$, consider $\Omega = (\Omega_1, \cdots, \Omega_{|\Omega|})$ and the partition of $(1, \cdots, N - 1)$ consisting of the following $|\Omega| + 1$ intervals:

$$(1, \cdots, \Omega_1 - 1), (\Omega_1, \cdots, \Omega_2 - 1), \cdots, (\Omega_{|\Omega|}, \cdots, N - 1) \tag{20}$$

Element $(1, \cdots, \Omega_1 - 1)$ is an interval of the kind defined in Lemma V.2 (symbol Δ), while all the remaining elements are intervals of the type defined in Lemma V.5 (symbol $\tilde{\Delta}$). In both kinds of intervals, there can be a maximum of one alarm, from which the proof can be derived. □

Under Theorem V.6, it is possible to construct Ω to ensure an upper limit on the number of alarms triggered by the solution of an instance $\mathcal{I}$. Notably, the outcome of Theorem V.6 is independent of the CUSUM sum value when the attack begins, S_1. The S value, not being transmitted over the network, could be unknown to a less powerful attacker. A lack of knowledge regarding S_1 could prevent an attacker from maintaining stealthiness in a zero-alarm scenario. However, in a multi-alarm scenario, the knowledge about the value of S_1 is less consequential. If $S^*_1 \geq S_1$ (for instance, both are zero), we have $|\mathbf{C}|^{=1} \leq |\Omega|$, since no alarm can occur in the initial time interval $(1, \cdots, \Omega_1 - 1)$ according to Lemma V.3. In contrast, if $S^*_1 < S_1$, then $|\mathbf{C}|^{=1} \leq |\Omega| + 1$, as per Theorem V.6. Hence, in a multi-alarm scenario, ignorance of the value of S_1 could potentially increase the number of alarms by just one, emphasizing the increased risk in multi-alarm scenarios compared to single-alarm ones.

Lemma V.7. *For all possible Ω in an instance $\mathcal{I}$, if an additional index element is included in Ω, the impact cannot decrease.*

Proof. Because $S^*_i, z^*_i, b \geq 0$, constraint 8p of Problem 1 is more relaxed than constraint 8n. Therefore, including an additional index to Ω relaxes the problem and can improve the solution (i.e., the impact). □

As per Lemma V.7, an instance $\mathcal{I}$ with $|\Omega| > 0$ (multi-alarm) is more relaxed than a modified instance where $|\Omega| = 0$ (zero-alarm). Consequently, the impact of a single-alarm solution is less than or equal to the impact of any possible multi-alarm solution.

When acceptable transparency implies a high probability of alarms (short ARL), the WCI assessment should be performed against multi-alarm attacks (response to RQ1). The reason is that from Lemma V.7, performing WCI assessment by considering $\mathcal{I}$ with $|\Omega| = 0$ (zero-alarm) may underestimate the actual deviation that $|\Omega| > 0$ (multi-alarm) could achieve. To answer RQ2, this paper proposes a new method to find attacks that produce alarms at a rate substantially equal to ARL^0. Hence they are hidden to the detector defined in Section V.

We propose to find multi-alarm attacks guaranteeing an upper-bound on the number of alarms by using an instance of Problem 1 where

$$|\Omega| = |\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^{\text{D}} - 1 \tag{21}$$

so that, for Theorem V.6, $|\mathbf{C}|^{=1} \leq |\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^{\text{D}}$ and Eq. (15) is True.

One possible criterion for the formation of Ω is to space the presence of constraints Eq. (8p) at a distance of $|\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^D$ samples, so that in the absence of a priori information we expect alarms to occur at the same distance. We highlight that the proposed way to form Ω is just one among the $\binom{N}{|\Omega|}$ possible, which suggests how finding a global optimum in the multi-alarm case is inherently more complex than the zero-alarm case.

The proposed $\mathcal{I}$ has Ω such that

$$\{i \in \Omega \mid \mathrm{mod}(i, |\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^D) = 0\}, i \in (1, \cdots, N - 1) \tag{22}$$

where mod is the modulus operator. Eq. (22) determines $|\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^D$ intervals as in Eq. (20) i.e., $\Omega = \big(|\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^D, 2(|\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^D), \cdots\big)$. For example, if $\delta^D = 1$, $|\mathbf{C}|^{=1}_{\text{TARGET}} = 30$, and $M = 330$ samples, then elements in Ω has a spacing of 31 i.e., $\Omega = (31, 62, 93, \cdots, 310)$.

B. Lower bound on the number of alarms

The optimization problem defined in section V-A guarantees an upper bound on the number of alarms, but there are no guarantees on the minimum number of alarms, which could lead to detection for Eq. (16). On the one hand, it is intuitive that maximizing the deviation tends to increase the anomaly, hence, producing alarms whenever possible (one alarm in each interval determined by Ω). On the other hand, it is not said that the optimal solution will generate the maximum number of alarms $|\Omega| + 1$. The actuators' saturation prevents an unbounded energy rate from being injected into the system. Consequently, there can be time intervals in which providing more anomalies is not useful to increase the deviation.

Since the attacker can always check the effect of an attack before injecting it into the real system, (s)he can adjust Ω accordingly to the number of alarms $|\mathbf{C}|^{=1}$, as obtained through Eq. (18). The Algorithm 1 details at a high level of abstraction the process of finding an attack sequence that delivers several alarms in a prescribed range.

Algorithm 1 Sub-optimal hidden multi-alarm attack sequence

1: **Input:** $\mathcal{I}$, $|\mathbf{C}|^{=1}_{\text{TARGET}}$, δ^D
2: `numelC` $\leftarrow |\mathbf{C}|^{=1}_{\text{TARGET}} + \delta^{\text{D}} - 1$
3: Initialize Ω based on `numelC`
4: Form instance $\mathcal{I}$ based on Ω
5: Solve $\mathcal{I}$ to obtain $\mathcal{X}^*$
6: Obtain $\mathbf{C}$ from $\mathcal{X}^*$ and function $\mathcal{E}$
7: compute `low` $= -|\mathbf{C}|^{=1} + |\mathbf{C}|^{=1}_{\text{TARGET}} - \delta^D$
8: compute `up` $= |\mathbf{C}|^{=1} - |\mathbf{C}|^{=1}_{\text{TARGET}} - \delta^D$
9: **while** (`low` > 0 OR `up` > 0) **do**
10: Update `numelC`
11: Form Ω based on `numelC`
12: Repeat lines $4-8$ updating $\mathbf{C}$
13: **end while**
14: **Output:** attack sequence a^* from $\mathcal{X}^*$

In the Algorithm, at line 2, the variable `numelC` is initialized so that the first time an attack sequence is obtained, it will not exceed the prescribed maximum number of alarms. At line 3, the attacker could equally distribute constraints of Eq. (8p) as in Eq. (22). At line 9, the algorithm ensures that the attack sequence remains hidden. When line 9 is visited for the first time, the *while* loop is only entered if not enough alarms are produced. At line 11, the value of Ω is modified, which may increase or decrease the number of alarms. There exists at least one criterion to change Ω such that an infinite loop is prevented and a hidden attack is found, as explained by Lemma V.8.

Lemma V.8. *Consider a generic Ω such that $|\mathbf{C}|^{=1} < |\mathbf{C}|^{=1}_{TARGET} - \delta^D$ (meaning, there are insufficient alarms for the attack to remain hidden). Assume $\Omega = (1, \cdots, N-1) \implies |\mathbf{C}|^{=1} \geq |\mathbf{C}|^{=1}_{TARGET} - \delta^D$. Then, by progressively incorporating new index elements into Ω following any criteria, it becomes possible to eventually find an attack sequence where $|\mathbf{C}|^{=1} = |\mathbf{C}|^{=1}_{TARGET} - \delta^D$ (thus the attack is hidden), and this holds for all δ. Moreover, every time a new index element is incorporated into Ω, the value of the optimal solution (or the impact) cannot decrease.*

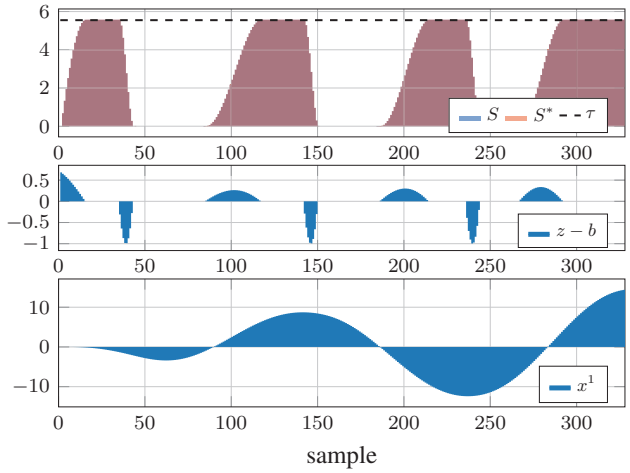

Fig. 3: Results for an optimal zero-alarm attack on a mass-spring-damper system, with $u_{\max} = -u_{\min} = 4$.

Proof. The assumption is justifiable given that if $\Omega = (1, \cdots, N-1)$, the corresponding optimization problem becomes sufficiently relaxed such that a solution may potentially generate an alarm sequence exhibiting $\text{ARL} = 1$. As per the results in Section V, including a new index element in Ω increases by one the number of intervals as defined in Theorem V.6, hence it might generate an additional alarm. As per our assumption, under any criteria by which a new index is iteratively incorporated into Ω, there must exist an iteration before reaching $\Omega = (1, \cdots, N-1)$ where $|\mathbf{C}|^{=1} = |\mathbf{C}|^{=1}_{\text{TARGET}} - \delta^{\text{D}}$. Furthermore, whenever an additional index is incorporated, the problem becomes more relaxed as per Lemma V.7; hence the impact cannot decrease. □

VI. Simulated results

This section presents the results of simulated attacks on a mass-spring-damper system, particularly on the first component x^1.

A. Zero vs multi alarm

We tune a CUSUM ADS to yield $\text{ARL}^0 = 30$. Per Eq. (6), this corresponds to a CUSUM threshold, τ, of 5.44. The higher-level ADS operates with an observation window of $M = 330$ and an alarm count tolerance of $\delta^D = 1$. As per Eq. (14), the target CUSUM output, $|\mathbf{C}|^{=1}_{\text{TARGET}}$, is set at 11. For Eqs. (15) and (16), an attack of duration $N = M$ is hidden if it results in CUSUM alarm count, $|\mathbf{C}|^{=1}$, within the range of [10,12].

The graph presented in Fig. 3 pertains to an optimal zero-alarm attack, which, given $|\mathbf{C}|^{=1} = 0$, is not hidden. In this

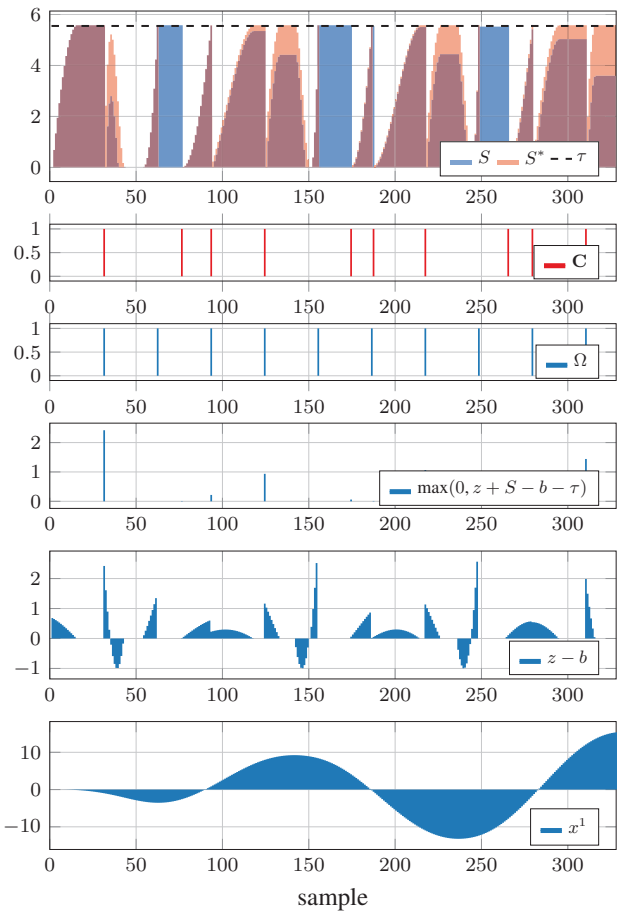

Fig. 4: Results for an optimal multi-alarm attack on a mass-spring-damper system, with $u_{\max} = -u_{\min} = 4$.

context, $z-b$ illustrates the incremental change of the CUSUM sum over time. In the figure, we juxtapose the evolution of the CUSUM sum in the optimal solution, denoted as S^*, and its counterpart in the actual ADS, denoted as S. The data is obtained through Eq. (18) which consists in applying a^* to the recursive equations, as given in Eq. (1)–(4). In this scenario, S and S^* coincide and do not breach the CUSUM threshold, τ. As a result of this attack, there is a final deviation in the first Plant's state component x_N^1 of 14.4.

Fig. 4 represents an optimal multi-alarm attack. In accordance with Eq. (21), we set $|\Omega|^{=1} = 10$, so that invoking Theorem V.6, $|\mathbf{C}|^{=1} \leq |\mathbf{C}|^{=1}_{\text{TARGET}} = 11$ alarms is guaranteed. Moreover, $|\mathbf{C}|^{=1} \leq 10$, as in the simulations, we enforce the condition $S_1 = S_1^* = 0$ (refer to the discussion on Theorem V.6 for further details). Elements of Ω are distributed as in Eq. (22) consisting of spacing between elements of 31 samples. The evolution of S and S^* can be observed in the figure, which aligns with the results discussed in Section V. The attack induces $|\mathbf{C}|^{=1} = 10$ alarms. The term $\max(0, z + S - b - \tau)$ conveys the extent to which S exceeds

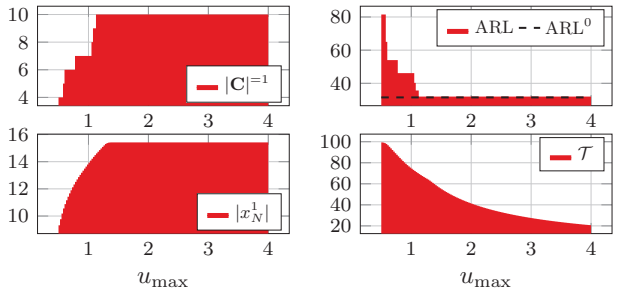

Fig. 5: Results from a series of optimal multi-alarm attacks as the saturation value $u_{\max} = -u_{\min}$ increases, and Ω has a spacing between elements of 31 samples.

τ at the instant an alarm is produced. Note that such quantity is not visible in the evolution of S because, according to Eq. (4), it is discarded upon resetting S. The final deviation of the attack is measured at 15.45. Compared to the zero-alarm attack, not only the multi-alarm attack is hidden, but it improves the impact of $\sim 8\%$.

B. *Influence of actuators saturation value*

This section examines a multi-alarm scenario wherein the actuator's saturation is reduced relative to the case outlined in Section VI-A. Specifically, the value of $u_{\max} = -u_{\min}$ is diminished from 4 to a range within $(0.5, 4)$.

Fig. 5 shows that attaining ten alarms is only possible when the actuator saturation exceeds $1.2N$. For smaller saturation values, the observed ARL is too large to determine a hidden attack. However, as introduced in Lemma V.8 and further detailed in the remainder of this section, it is always possible to produce more alarms (hence decrease the ARL) while at the same time not decreasing the impact. Fig. 5 also shows that for the current tuning (fixed to $\text{ARL}^0 = 30$), greater values of $u_{\max}$ render the ADS more effective in terms of transparency. In particular, for small values of $u_{\max}$, the transparency nearly reaches its maximum, informing that the ADS is ineffective, regardless of the impact. The figure illustrates that within the range $u_{\max} = (0.5, 1.2)$, saturation level is a limiting factor for both the number of alarms and the impact $|x_N^1|$. Beyond $u_{\max} = 1.2$, the maximum theoretical number of alarms of 10 is achieved, after which the impact remains constant (the only limiting factor becomes the prescribed constant tuning of the ADS). Beyond $u_{\max} = 1.2$, the transparency, defined in Eq. (11), continues to diminish because the ADS does not constrain $\mathcal{X}^{\text{U}}$.

The subsequent analysis is focused on the scenario where $u_{\max} = -u_{\min} = 0.65$. As demonstrated by Fig. 5 when Ω has a spacing between elements of 31 samples, we have $|\mathbf{C}|^{=1} = 6 \notin [10, 12]$, implying that the attack does not qualify as a hidden. Fig. 6 details the alarm times and the considered set Ω. According to Algorithm 1, in this case, Ω must be adjusted to generate sufficient alarms. Although Lemma V.8 guarantees a method for eventually identifying a hidden attack, we experiment with a different criterion that nonetheless yields

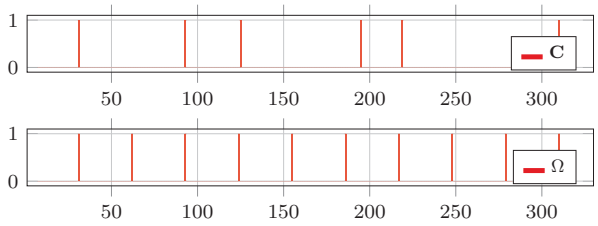

Fig. 6: Alarm sequence obtained with $u_{\max} = 0.65$ and Ω having a spacing between elements of 31 samples, determining 6 alarms.

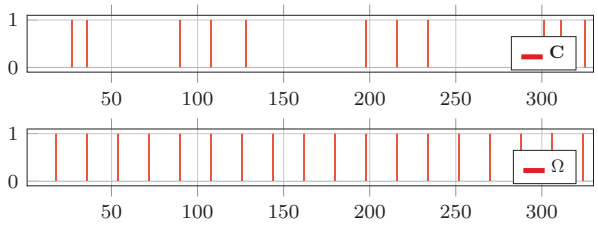

Fig. 7: Alarm sequence obtained with $u_{\max} = 0.65$ and Ω having a spacing between elements of 18 samples, determining 11 alarms (hence hidden).

a valid solution. Specifically, we iteratively solve multiple optimization problems, progressively increasing $|\Omega|$ following the criterion of Eq. (22). Table I shows the number of alarms obtained as the spacing between elements in Ω decreases. Any spacing in the range $(18, \dots, 14)$ determines a hidden attack. For example, with a spacing of 18 samples, we successfully generate 11 alarms, as further illustrated in Fig. 7.

VII. Conclusion

This paper introduces a new formulation for multi-alarm attacks on cyber-physical control systems secured by CUSUM-based anomaly detection schemes. This is proposed within the scope of a Worst Case Impact assessment. We have introduced the concept of transparency for an Anomaly Detection System (ADS), a measure of how much the presence of an ADS can limit the impact (Plant's state deviation) of stealth attacks. Through simulations on a mass-spring system, we show that to achieve acceptable transparency, the Average Run Length (ARL, which is the ratio between the number of samples in an observation window and the number of alarms) must be short. Consequently, with short ARLs, CUSUM alarms are triggered frequently, demanding a higher-level detection system to monitor the CUSUM ARL. Both theoretical analysis and simulation experiments indicate that multi-alarm attacks could result in a more significant state deviation than zero-alarm attacks. Additionally, knowing the CUSUM sum at the start of an attack does not significantly contribute to preserving stealthiness. This leads us to conclude that in the presence of short ARL, Worst Case Impact assessments should focus on multi-alarm attacks instead of zero-alarm attacks. We have formulated multi-alarm attacks by adjusting the convex constraints that model the discrete dynamics of the CUSUM, offering formal guarantees on the maximum number of alarms. We have proposed an algorithm that is guaranteed to identify multi-alarm attacks that remain hidden with respect to a higher-level detection system monitoring the CUSUM ARL in a time window. However, it is essential to acknowledge that our solution does not assure global optimality. The constraints in updating the CUSUM could be adjusted at different timings from those considered in this study, possibly leading to more dangerous attacks. Hence, the issue of Worst Case Impact assessment for short ARL is still an open question and requires further exploration.

Ω **spacing**	31	30	29	28	27	26	25	24	23
$\lvert\mathbf{C}\rvert^{=1}$	6	5	7	6	7	9	7	7	8
Ω **spacing**	22	21	20	19	**18**	**17**	**16**	**15**	**14**
$\lvert\mathbf{C}\rvert^{=1}$	8	9	8	9	**11**	**11**	**12**	**11**	**12**
Ω **spacing**	13	12	11	10	9	8	7	6	5
$\lvert\mathbf{C}\rvert^{=1}$	12	15	16	17	18	19	24	22	26

TABLE I: Number of alarms, $|\mathbf{C}|^{=1}$, obtained with different spacing between elements in Ω, in the case where $u_{\max} = -u_{\min} = 0.65N$.

References

[1] A. Khraisat *et al.*, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, 2019.

[2] T. K. Das, S. Adepu, and J. Zhou, "Anomaly detection in industrial control systems using logical analysis of data," *Computers & Security*, vol. 96, p. 101935, 2020.

[3] A. Teixeira *et al.*, "A secure control framework for resource-limited adversaries," *Automatica*, vol. 51, pp. 135–148, 2015.

[4] A. Teixeira, K. C. Sou, H. Sandberg, and K. H. Johansson, "Quantifying cyber-security for networked control systems," in *Control of cyber-physical systems*. Springer, 2013, pp. 123–142.

[5] J. Milošević, D. Umsonst, H. Sandberg, and K. H. Johansson, "Quantifying the impact of cyber-attack strategies for control systems equipped with an anomaly detector," in *ECC*, 2018, pp. 331–337.

[6] F. J. R. Lera, C. F. Llamas, Á. M. Guerrero, and V. M. Olivera, "Cybersecurity of robotics and autonomous systems: Privacy and safety," *Robotics-legal, ethical and socioeconomic impacts*, 2017.

[7] H. Olufowobi, U. Ezeobi, E. Muhati, G. Robinson, C. Young, J. Zambreno, and G. Bloom, "Anomaly detection approach using adaptive cumulative sum algorithm for controller area network," in *Proceedings of the ACM Workshop on Automotive Cybersecurity*, 2019, pp. 25–30.

[8] D. Umsonst, H. Sandberg, and A. A. Cárdenas, "Security analysis of control system anomaly detectors," in *ACC*, 2017, pp. 5500–5506.

[9] D. Umsonst, N. Hashemi, H. Sandberg, and J. Ruths, "Practical detectors to identify worst-case attacks," in *CCTA*, 2022, pp. 197–204.

[10] D. I. Urbina *et al.*, "Limiting the impact of stealthy attacks on industrial control systems," in *ACM SIGSAC Conf. on Computer and Comm. Security*, 2016, pp. 1092–1105.

[11] C. Murguia and J. Ruths, "Characterization of a cusum model-based sensor attack detector," in *CDC*, 2016, pp. 1303–1309.

[12] ——, "Cusum and chi-squared attack detection of compromised sensors," in *IEEE Conf. Contr. Appl.*, 2016, pp. 474–480.

[13] N. Hashemi, C. Murguia, and J. Ruths, "A comparison of stealthy sensor attacks on control systems," in *ACC*, 2018, pp. 973–979.

[14] L. F. Cómbita, A. A. Cárdenas, and N. Quijano, "Mitigation of sensor attacks on legacy industrial control systems," in *CCAC*, 2017.

[15] A. A. Cárdenas *et al.*, "Attacks against process control systems: risk assessment, detection, and response," in *ACM Symp. on Information, Computer and Comm. Security*, 2011, pp. 355–366.

[16] G. Gualandi, M. Maggio, and A. V. Papadopoulos, "Optimization-based attack against control systems with cusum-based anomaly detection," in *Med. Conf. Control and Aut. (MED)*, 2022, pp. 896–901.

[17] D. D. Nguyen, M. T. Le, and T. L. Cung, "The ability to detect the linear attack of wl-cusum and fma algorithms," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 1, pp. 131–140, 2023.