A Switching Multiplicative Watermarking Scheme for Detection of Stealthy Cyber-Attacks

Ferrari, Riccardo M.G.; Teixeira, Andre M.H.

DOI
10.1109/TAC.2020.3013850

Publication date
2021

Document Version
Final published version

Published in
IEEE Transactions on Automatic Control

Citation (APA)
Ferrari, R. M. G., & Teixeira, A. M. H. (2021). A Switching Multiplicative Watermarking Scheme for Detection of Stealthy Cyber-Attacks. *IEEE Transactions on Automatic Control*, *66*(6), 2558-2573. https://doi.org/10.1109/TAC.2020.3013850

A Switching Multiplicative Watermarking Scheme for Detection of Stealthy Cyber-Attacks

Riccardo M. G. Ferrari and André M. H. Teixeira

***Abstract*—This article addresses the detection of stealthy attacks on sensor measurements. Inspired in authentication schemes with weak cryptographic guarantees, we propose a watermarking approach to validate the data and its source. In particular, we propose a multiplicative scheme, where the sensor outputs are watermarked by a bank of filters, then transmitted through the possibly unsecured communication network. The original measurement data are finally reconstructed by a watermark remover. To allow the detection of replay attacks, the watermarking filters are devised as hybrid switching systems, whose parameters are assumed to be unknown to the adversary. Design rules are provided, guaranteeing that the nominal closed-loop performance is not deteriorated by the watermarking scheme and ensuring robust stability with mismatched filter parameters. Moreover, we design a switching protocol with no communication overhead to allow the watermarking filters to synchronously update their parameters. The detectability properties of cyber-attacks are analyzed, and the results are illustrated through numerical examples for replay and data injection attacks.**

***Index Terms*—Digital filters, equalizers, intrusion detection, security, watermarking.**

I. Introduction

CYBERSECURITY has become an increasingly important aspect of control systems in recent years, driven by the pervasive use of information technologies, as well as by the steadily increasing number of newly discovered vulnerabilities [1], [2] and of reported cyber-attacks [3].

An overview of existing cyber-threats and vulnerabilities in networked control systems is presented in [4]–[6]. Rational adversary models are highlighted as one of the key items in security for control systems, thus making adversaries endowed with intelligence and intent, as opposed to faults. Therefore, these adversaries may exploit existing vulnerabilities and limitations in the traditional anomaly detection mechanisms and remain undetected. In fact, Pasqualetti *et al.*[7] used such fundamental limitations to characterize a set of stealthy attack policies for networked systems modeled by differential-algebraic equations. Related stealthy attack policies were also considered in [6] and [8].

Detectability conditions of stealthy false-data injection attacks to control systems are examined in [9], where it is shown that they may become detectable due to mismatches between the system's and the attack's initial conditions. Additionally, modifications to the system dynamics that reveal stealthy attacks were also characterized. Recently, [10] proposed a static output coding scheme combining the outputs of multiple sensors to reveal stealthy data injection attacks on sensors.

However, both approaches present certain limitations. On the one hand, the plant's initial conditions cannot be directly controlled, and changing the system dynamics may negatively affect performance. On the other hand, sensor coding schemes require additional communication between sensors and the controller, and it would not be applicable in single-output systems. These limitations can be tackled by using a multiplicative watermarking scheme, as discussed in this article.

Watermarking is a well-known solution to the problem of authenticity and integrity verification in the field of multimedia data [11]. An additive watermarking scheme has been proposed by [12] and by [13] to detect replay attacks, where noise is purposely injected in the system by the actuators to watermark the sensor outputs through known correlations. A similar, but distributed, approach was recently proposed to detect replay attacks in interconnected microgrids [14]. However, this scheme decreases the performance of the system and fails to detect additive stealthy attacks, drawbacks that can be tackled by employing multiplicative watermarks.

Recently, Weerakkody and Sinopoli [15] proposed the use of an external auxiliary system, with time-varying dynamics unknown to the adversary, whose output is transmitted to the anomaly detector and used to detect the presence of integrity attacks. While sharing similarities with our proposed multiplicative watermarking, the approach in [15] imposes further burdens on the system, such as the communication of the external system's measurement signals and the use of an additional state estimator, which are not required in our watermarking solution.

Manuscript received December 25, 2019; revised June 23, 2020; accepted July 14, 2020. Date of publication August 4, 2020; date of current version May 27, 2021. This work was supported in part by the European Union Seventh Framework Programme (FP7/2007-2013) under Grant 608224, the EU H2020 Programme under Grant 707546 (SURE), the Swedish Research Council under Grant 2018-04396, and the Swedish Foundation for Strategic Research. Recommended by Guest Editor Prof. Francesco Bullo. *(Corresponding author: André M. H. Teixeira.)*

Riccardo M. G. Ferrari is with the Delft Center for Systems and Controls, Delft Technical University, 2628, CD Delft, The Netherlands (e-mail: riccardo.mg.ferrari@gmail.com).

André M. H. Teixeira is with the Division of Signals and Systems, Department of Electrical Engineering, Uppsala University, 751 03 Uppsala, Sweden (e-mail: andre.teixeira@angstrom.uu.se).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2020.3013850

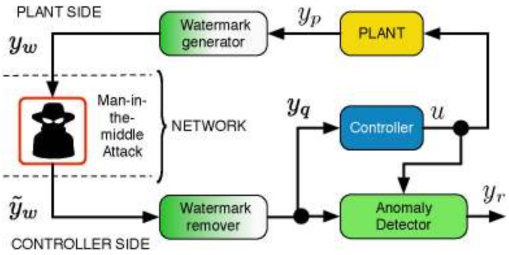

Fig. 1. Scheme of the proposed watermarking scheme under MITM attack.

As main contributions of this article, we consider the modular multiplicative watermarking scheme recently proposed in [16]–[18] against cyber-attacks, where the sensor outputs are watermarked by being fed to a watermark generator, and the watermark is later removed at the controller, therefore, not requiring communication between multiple sensors and ensuring a modular architecture.

As illustrated in Fig. 1, the proposed watermarking solution resembles a channel encryption scheme. Indeed, one may view watermarking as mechanisms to enforce authentication of the data and its source, generally with weaker cryptographic guarantees than strong message encryption schemes. On the one hand, this translates into lighter computational requirements and therefore smaller delays, although at the cost of more easily breakable confidentiality of the communicated data. On the other hand, watermarking still provides a feasible approach to ensure authentication, by allowing the detection of eventual corruption of the data and its watermark by adversaries. In networked control systems, where meeting real-time constraints is critical, and authentication and data integrity are typically more important than data confidentiality [19], the use of strong cryptographic methods may be an overdimensioned solution with several practical limitations. In contrast, multiplicative watermarking provides a feasible light-weight alternative to authenticate the data.

In the proposed watermarking scheme, the generator processes the measurements and transmits the watermarked data, which is then received and processed by the remover to reconstruct the original measurements. The rationale for including the proposed watermarking scheme is to make man-in-the-middle (MITM) attacks detectable, by having them cause an imperfect reconstruction of the plant output, a condition that will cause a detection by the *anomaly detector* [16]–[18]. Moreover, by carefully designing the watermark generator and remover as hybrid switching systems with piece-wise linear dynamics, while ensuring the perfect reconstruction of the plant outputs, we successfully introduce time-varying properties on the communicated data that facilitate the detection of replay attacks.

Given the advantages of multiplicative watermarking over classical encryption, and its ability to reveal stealthy attacks as illustrated in [16]–[18], this article addresses the design of the watermark filters. In particular, we show how the watermarking scheme can be designed to detect cyber-attacks, without affecting the performance of the system in the absence of attacks. The design guidelines of the watermarking filters are independent of the anomaly detection and control schemes, thus ensuring modularity. Moreover, we propose a synchronization protocol between the hybrid switching watermark generator and remover filters, so that both filters update their parameters simultaneously. Stability of the closed-loop system with the proposed watermarking scheme is also analyzed, including for the case of constant but mismatched parameter filters at the generator and remover. Finally, we investigate detectability guarantees provided by the scheme.

The outline of this article is as follows. In Section II, we describe the problem formulation, as well as a generic man-in-the-middle attack scenario and recall instances of attacks that are undetectable without watermarking. A first description of the closed-loop system with watermarking filters is also provided. The design of the sensor watermarking scheme is addressed in Section III, where design guidelines for the watermarking scheme are provided, together with an introductory description and illustrative example of the switching protocol for updating the watermarking parameters. A more generic switching protocol to ensure the synchronous update of the watermarking parameters is designed in Section IV. Section V analyzes the stability of the closed-loop system with the proposed watermarking scheme. Detectability properties are investigated in Section VI, while numerical results illustrating the effectiveness of the proposed solutions are reported in Section VII. Section VIII concludes this article with final remarks and possible future work.

II. Problem Formulation

In this section, we present the networked control system that is the target of so-called MITM cyber-attacks. Different instances of MITM attacks are described, in particular replay attacks, which are further addressed by this present article. The main elements of our proposed solution are also introduced, namely an attack-detection scheme based on switching multiplicative watermarks.

The modeling framework described in [6] and in [16]–[18] will be considered, where the control system is composed by a physical plant ($\mathcal{P}$) and a feedback controller ($\mathcal{C}$), interconnected via a communication network. While the communication network in general can be used to convey both measurements of the plant output to the controller, and control actions to the plant, without loss of generality in this article, we will focus only on the communication of the plant outputs and on cyber-attacks affecting such communication (see Fig. 1).

A. Networked Control System

The physical plant and controller are modeled in a discrete-time state-space form as, respectively,

$$\mathcal{P}:\begin{cases} x_p[k+1] = A_p x_p[k] + B_p u[k] + \eta[k] \\ y_p[k] = C_p x_p[k] + \xi[k] \end{cases}$$

$$\mathcal{C}:\begin{cases} x_c[k+1] = A_c x_c[k] + B_c y_q[k] \\ u[k] = C_c x_c[k] + D_c y_q[k] \end{cases} \tag{1}$$

where $x_p[k] \in \mathbb{R}^{n_p}$ and $x_c[k] \in \mathbb{R}^{n_c}$ are the state variables, $u[k] \in \mathbb{R}^{n_u}$ is the vector of control actions applied to the plant, $y_p[k] \in \mathbb{R}^{n_y}$ is the vector of plant outputs, $y_w[k] \in \mathbb{R}^{n_y}$ is the vector of watermarked measurements transmitted by the sensors, and $\tilde{y}_w[k] \in \mathbb{R}^{n_y}$ is the watermarked data received at the controller's side, which is possibly different than y_w due to the presence of a MITM adversary. At the controller's side, the watermarked data are processed through a watermark remover, which produces $y_q[k] \in \mathbb{R}^{n_y}$ that is fed to the controller and anomaly detector. Finally, $\eta[k]$ and $\xi[k]$ denote the unknown process and measurement disturbances, respectively.

Assumption 1: The uncertainties represented by η and ξ are unknown, but their norms are upper bounded by some known and bounded sequences $\bar{\eta}[k]$ and $\bar{\xi}[k]$.

The anomaly detector ($\mathcal{R}$) is collocated with the controller and it evaluates the behavior of the plant based only on the open-loop plant models and the available input and output data $u[k]$ and $y_q[k]$. It is described by the following equation in discrete-time state-space form

$$\mathcal{R}: \begin{cases} x_r[k+1] = A_r x_r[k] + B_r u[k] + K_r y_q[k] \\ \qquad y_r[k] = C_r x_r[k] + D_r u[k] + E_r y_q[k] \end{cases} \tag{2}$$

where $x_r \in \mathbb{R}_p^n$ is the detector's state vector and $y_r \in \mathbb{R}_y^n$ its output vector, also called *residual*.

Definition 1: Given the residue signal y_r, an attack is *detected* at a time instant k if

$$|y_{r,(i)[k]}| \geq \bar{y}_{r,(i)}[k] \tag{3}$$

for at least one component $i \in \{1, \ldots, n_y\}$, where $\bar{y}_r[k] \in \mathbb{R}_+^{n_y}$ is a robust time-varying *detection threshold*.

The main focus of this article is to investigate the detection of MITM attacks on sensors. This attack scenario, as well as a fundamental limitation in their detectability akin to the results of [6], [7], are described next, where the detectability of attacks is discussed according to the following definition.

Definition 2: Suppose that the closed-loop system is at equilibrium such that $y_r[-1] = 0$, and that there are no unknown disturbances, i.e., $\eta[k] = 0$ and $\xi[k] = 0$ for all k. An anomaly occurring at $k = k_a \geq 0$ is said to be ε-stealthy if $\|y_r[k]\|_\infty \leq \varepsilon$ for all $k \geq k_a$.

In particular, an ε-stealthy anomaly is termed as simply *stealthy*, whereas a 0-stealthy anomaly is named *undetectable*.

B. MITM Attacks

Next, we briefly describe the main assumptions regarding the adversary's capabilities considered in this article.

In the present scenario, a malicious adversary is able to access and corrupt the watermarked measurements sent by the sensors to the controller, which is captured by the equation

$$\tilde{y}_w[k] = \phi\left(Y_{w,(k-\tilde{N},k]}\right) \tag{4}$$

where $Y_{w,(k-\tilde{N},k]} \triangleq [y_w[k-\tilde{N}+1] \ldots y_w[k]] \in \mathbb{R}^{n_y} \times \mathbb{R}^{\tilde{N}}$ is a data matrix containing the last $\tilde{N}$ values of the watermarked measurements y_w, and $\phi : \mathbb{R}^{n_y} \times \mathbb{R}^{\tilde{N}} \mapsto \mathbb{R}^{n_y}$ is a mapping describing the attacker policy for corrupting the data. Note that this may include false-data injection attacks $\tilde{y}_w[k] = y_w[k] + a[k]$, where malicious data $a[k]$ are added to the measurement [16], replay attacks $\tilde{y}_w[k] = y_w[k-T]$ [17], and rerouting attacks $\tilde{y}_w[k] = R y_w[k]$, where R is a routing matrix [18].

Adversaries with the following characterizations are considered in this present article.

Attack Goals and Constraints: The adversary aims at disrupting the system's behavior by corrupting the sensor data, while remaining stealthy (see Definition 2).

Disruption and Disclosure Resources: The adversary is assumed to have disruption resources to corrupt the measurement data, as well as disclosure resources to eavesdrop on the transmitted data.

Model Knowledge: In the present scenario, the adversary also has access to the detailed nominal model of the plant, (A_p, B_p, C_p), which may be used to compute the attack policy.

As mentioned in Section I, in common systems without measurement watermarking, i.e., $y_w[k] = y_p[k]$ and $y_q[k] = \tilde{y}_p[k]$, there are several instances of MITM attacks that remain stealthy with respect to arbitrary passive linear time-invariant (LTI) anomaly detectors. See for instance [7] for false-data injection attacks, and [13] for replay attacks. Moreover, additive watermarking techniques as proposed in [13] have the caveats of not facilitating the detection of additive attacks, and of perturbing the nominal system operation and degrading performance in the absence of attacks. To tackle these issues and allow for the detectability of generic MITM attacks, we propose the use of the multiplicative watermarking scheme illustrated in Fig. 1, and further described as follows.

C. Watermarking–Based Anomaly Detection Scheme

To detect the presence of MITM attacks, we propose in this article to leverage three specific blocks of the networked control system, as outlined in Fig. 1: a *Watermark Generator* $\mathcal{W}$, a *Watermark Remover* $\mathcal{Q}$, and an *Anomaly Detector* $\mathcal{R}$.

The watermark generator and remover are hybrid discrete-time linear systems whose dynamics between switches are described by the following state-space equations:

$$\begin{aligned} &\mathcal{W}: \begin{cases} x_w[k+1] = A_w(\theta_w[k]) x_w[k] + B_w(\theta_w[k]) y_p[k] \\ \qquad y_w[k] = C_w(\theta_w[k]) x_w[k] + D_w(\theta_w[k]) y_p[k] \end{cases} \\ &\mathcal{Q}: \begin{cases} x_q[k+1] = A_q(\theta_q[k]) x_q[k] + B_q(\theta_q[k]) y_w[k] \\ \qquad y_q[k] = C_q(\theta_q[k]) x_q[k] + D_q(\theta_q[k]) y_w[k] \end{cases} \end{aligned} \tag{5}$$

where the vectors $x_w, x_q \in \mathbb{R}^{n_w}$, and $y_w, y_q \in \mathbb{R}^{n_y}$ represent, respectively, the state of the watermark generator $\mathcal{W}$ and of the watermark remover $\mathcal{Q}$ and their outputs. The vectors θ_w and $\theta_q \in \mathbb{R}^{n_\theta}$ denote piece-wise constant parameters affecting the dynamics of $\mathcal{W}$ and $\mathcal{Q}$. They are updated only at switching times, and the updates are described by

$$\begin{aligned} &\mathcal{W}: \begin{cases} \theta_w^+[k] = \sigma_w(\theta_w^-[k]) \\ x_w^+[k] = \rho_w(x_w^-[k], y_p[k], \theta_w^-[k], \theta_w^+[k]) \end{cases} \quad \text{if } \tau_w[k] = 1 \\ &\mathcal{Q}: \begin{cases} \theta_q^+[k] = \sigma_q(\theta_q^-[k]) \\ x_q^+[k] = \rho_q(x_q^-[k], y_w[k], \theta_q^-[k], \theta_q^+[k]) \end{cases} \quad \text{if } \tau_q[k] = 1 \end{aligned} \tag{6}$$

where the functions $\sigma_w, \sigma_q : \mathbb{R}^{n_\theta} \mapsto \mathbb{R}^{n_\theta}$ and $\rho_w, \rho_q : \mathbb{R}^{n_w} \times \mathbb{R}^{n_y} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_\theta} \mapsto \mathbb{R}^{n_w}$ denote, respectively, the *switching maps* of $\mathcal{W}$ and $\mathcal{Q}$ and their *jump maps*. By drawing on the hybrid systems literature [20], [21], we denote here the value of a variable after the switch has been applied by a superscript "+". Furthermore, in the present article, we introduce also a superscript "−" to denote values right before the switch.

Finally, we have the following definition of triggering functions.

Definition 3: The functions $\tau_w, \tau_q : \mathbb{R}^{n_y} \mapsto \{0, 1\}$ are said to be the *triggering functions* of $\mathcal{W}$ and $\mathcal{Q}$ if the *triggering sets* $\mathcal{C}_w \triangleq \{y_p : \tau_w(y_p) = 1\}$ and $\mathcal{C}_q \triangleq \{y_w : \tau_q(y_w) = 1\}$ are convex and open. Furthermore, the sequences $\mathcal{K}_w \triangleq \{\kappa_w : \tau_w(y_p[\kappa_w]) = 1\}$ and $\mathcal{K}_q \triangleq \{\kappa_q : \tau_q(y_w[\kappa_q]) = 1\}$ are the *switching time sequences* of, respectively, $\mathcal{W}$ and $\mathcal{Q}$.

The triggering functions, switching and jump maps will be characterized in Section III-B. Recalling the objective that the watermark remover is able to reconstruct the original measurements, we make the following assumption.

Assumption 2: The sequence of parameter vectors $\theta_w[k]$ and $\theta_q[k]$ generated by the switch functions σ_w and σ_q and the dependence of the matrices A_w, B_w, C_w, and D_w on θ_w and of the matrices A_q, B_q, C_q, and D_q on θ_q are such that, for every instant k

1) $\mathcal{W}$ is stable and invertible;
2) $\mathcal{Q}$ is stable;
3) $\theta_w = \theta_q \Rightarrow \mathcal{Q}$ is the inverse of $\mathcal{W}$.

Having defined all the elements illustrated in Fig. 1, we may now describe the full dynamics of the closed-loop system by having, at the plant's side, the plant $\mathcal{P}$ in cascade with the watermark generator $\mathcal{W}$.

The sensors transmit the watermarked data $y_w[k]$ to the controller's side, which may be corrupted by a MITM adversary as described in (25), being replaced by $\tilde{y}_w[k]$.

At the controller's side of the network, we have the watermark remover $\mathcal{Q}$ in cascade with the controller and detector. The received data $\tilde{y}_w[k]$ are fed to the watermark remover $\mathcal{Q}$, which produces $y_q[k]$. The remover's output is in turn used to compute the residual and control input as

$$\mathcal{F}_{cr} : \begin{cases} x_{cr}[k+1] = A_{cr} x_{cr}[k] + B_{cr} y_q[k] \\ y_r[k] = C_{cr} x_{cr}[k] + D_{cr} y_q[k] \\ u[k] = C_u x_{cr}[k] + D_u y_q[k] \end{cases} \tag{7}$$

where $x_{cr}[k] = [x_c[k]^\top \; x_r[k]^\top]^\top$, and the matrices A_{cr}, B_{cr}, C_{cr}, D_{cr}, C_u, and D_u are derived from (1).

Remark 1: The rationale for including the proposed *active* watermarking scheme is to make attacks detectable by having them cause an imperfect reconstruction of y_p, a condition that will cause a detection (cfr. Definition 1) by the *anomaly detector* described in Section VI. Indeed, in the absence of such a watermarking scheme, it can be shown that there exist classes of *stealthy* attacks that are not detectable by any passive LTI model-based anomaly detector [7], [13], [16]–[18].

Assumption 3: The initial values $x_w[0]$, $x_q[0]$ and $\theta_w[0]$, $\theta_q[0]$ and the functions ρ_w, ρ_q and σ_w, σ_q and τ_w, τ_q are not known to the adversary, but are a shared secret between the watermark generator $\mathcal{W}$ and the watermark remover $\mathcal{Q}$.

Given the aforementioned watermarking scheme, we are interested in designing the filter dynamics so that three objectives are met: 1) nominal performance is ensured without attack; 2) the system is robustly stable to nonsynchronized watermarking filters; 3) undetectable attacks policies with respect to the nominal systems become detectable with the proposed watermarking scheme. These objectives are the focus of the following three sections.

III. Design of the Watermarking Scheme

In this section, we address the design of the watermarking scheme as to guarantee that, without attacks, nominal performance is not affected. This is done in two steps: 1) the nominal performance of the closed-loop system is not affected by the watermarking scheme with matched filter parameters in between switching events; and 2) the watermarking scheme is able to trigger a simultaneous update of the parameter θ at the generator and remover without additional communications and without affecting performance.

In the remainder of this article, we assume that the filters are designed so that they are stable. For notation simplicity and without loss of generality, we consider the single sensor case, i.e., $n_y = 1$. Note that the results extend straightforwardly to the multiple sensor case.

A. Design for Performance Between Switching Events

To guarantee that nominal performance is not affected by the presence of the watermarking generator and remover, we must ensure that $y_p[k] = y_q[k]$ holds at all times. As we shall see next, three conditions are required for this, namely that the generator and remover use the same filter parameter, that their state-space dynamics are matched so that one is the inverse of the other, and that their states are also matched accordingly.

The following result provides relations between the matrices in (5), which guarantee that, for $\theta_w = \theta_q$, one filter is the inverse of the other.

Lemma 1: Consider the watermark generator $\mathcal{W}(\theta)$ and the watermark remover $\mathcal{Q}(\theta)$ using the same parameters, and let $\mathcal{W}(z;\, \theta) \triangleq C_w(zI_N - A_w)^{-1}B_w + D_w$ and $\mathcal{Q}(z;\, \theta) \triangleq C_q(zI_N - A_q)^{-1}B_q + D_q$ be the respective transfer functions. The equality $\mathcal{Q}(z;\, \theta)\mathcal{W}(z;\, \theta) = 1$ holds if, and only if, there exists an invertible matrix T satisfying the following relations:

$$D_q C_w + C_q T = 0, \quad T^{-1} B_q D_w = B_w, \quad D_q D_w = 1$$
$$T^{-1} A_q T + T^{-1} B_q C_w = T^{-1} A_q T - B_w C_q T = A_w. \tag{8}$$

Proof: The proof follows directly from the derivation of the inverse of a square system with invertible direct feed-through term and realization results [22]. ■

The next result ensures that nominal performance is ensured if the conditions of Lemma 1 hold and the states of the filters are matched at switching times, i.e., $x_w[\kappa_j] = x_q[\kappa_j]$ holds for all $\kappa_j \in \mathcal{K}_w$, with $j \in \mathbb{N}$ denoting the generic index of the switching times of $\mathcal{W}$.

Theorem 1: Consider the watermarking filters $\mathcal{W}(\theta)$ and $\mathcal{Q}(\tilde{\theta})$. The trajectories of the closed-loop system with and without the watermarking scheme are the same if, and only if, $\theta = \tilde{\theta}$, the relations in Lemma 1 are satisfied, and $x_w[\kappa_j] = x_q[\kappa_j]$ holds for all $\kappa_j \in \mathcal{K}_w$. Furthermore, if $x_w[\kappa_j] = x_q[\kappa_j]$ for $\kappa_j \in \mathcal{K}_w$ and $\theta = \tilde{\theta}$, then $x_w[k] = x_q[k]$ holds for all $k \in [\kappa_j,\ \kappa_{j+1})$.

Proof: The proof hinges on the fact that nominal performance hold is equivalent to have $y_p[k] = y_q[k]$ for all times k, and the proof follows by showing that the latter equality is ensured by the conditions stated in the theorem.

To do so, we shall consider the variables $x_{wq}[k] \triangleq x_w[k] - x_q[k]$ and $\Delta y_q[k] = y_q[k] - y_p[k]$. The trajectory of the variable $\Delta y_q[k]$ is described by the state-space equations

$$\mathcal{D}(\theta,\tilde{\theta}) : \begin{cases} \begin{bmatrix} x_w[k+1] \\ x_q[k+1] \end{bmatrix} = \begin{bmatrix} A_w(\theta) & 0 \\ B_q(\tilde{\theta})C_w(\theta) & A_q(\tilde{\theta}) \end{bmatrix} \begin{bmatrix} x_w[k] \\ x_q[k] \end{bmatrix} \\ \quad + \begin{bmatrix} B_w(\theta) \\ B_q(\tilde{\theta})D_w(\theta) \end{bmatrix} y_p[k] \\ \Delta y_q[k] = \begin{bmatrix} D_q(\tilde{\theta})C_w(\theta) & C_q(\tilde{\theta}) \end{bmatrix} \begin{bmatrix} x_w[k] \\ x_q[k] \end{bmatrix} \\ \quad + \left(D_q(\tilde{\theta})D_w(\theta) - I_{n_y} \right) y_p[k]. \end{cases} \tag{9}$$

Replacing $x_q[k]$ with $x_{wq}[k]$, having equal filter parameters, and inserting (8) yields

$$\begin{bmatrix} x_w[k+1] \\ x_{wq}[k+1] \end{bmatrix} = \begin{bmatrix} A_w & 0 \\ 0 & A_q \end{bmatrix} \begin{bmatrix} x_w[k] \\ x_{wq}[k] \end{bmatrix} + \begin{bmatrix} B_w \\ 0 \end{bmatrix} y_p[k]$$

$$\Delta y_q[k] = \begin{bmatrix} 0 & -C_q \end{bmatrix} \begin{bmatrix} x_w[k] \\ x_{wq}[k] \end{bmatrix}.$$

Note that having $x_w[k_j] = x_q[k_j]$ is equivalent to $x_{wq}[k_j] = 0$, which in turn ensures that $x_{wq}[k] = 0$ and $\Delta y_q[k] = 0$ for all $k \in [\kappa_j,\ \kappa_{j+1})$, which concludes the proof. ■

The above results guarantee that, under the watermarking scheme with matched filter parameters between switching events, suitable design choices can be made so that the trajectories of the closed-loop system are unaffected. This in turn ensures a separation principle in the design of the watermarking scheme and the feedback controller and anomaly detector.

B. Event-Triggered Watermark Switching Protocols

Following the result in Theorem 1, we would like $\mathcal{W}$ and $\mathcal{Q}$ to be synchronized at every time instant k. Synchronization over interswitching times is ensured by Theorem 1 if, for any $\theta_w = \theta_q$, the dynamical model of $\mathcal{Q}$ is the stable inverse of the one of $\mathcal{W}$ (*matched filters*, as characterized in Lemma 1). This amounts to choosing appropriate parameters θ_w, which can be designed offline.

As for synchronization at switching times $\kappa_w \in \mathcal{K}_w$ and $\kappa_q \in \mathcal{K}_q$, the following synchronization requirements must be fulfilled when designing $\mathcal{W}$ and $\mathcal{Q}$:

1) $\mathcal{K}_w = \mathcal{K}_q$ (*synchronized switch times*).
2) The outputs of their switch functions σ and jump functions ρ are the same (*synchronized switches and jumps*).
3) $y_q^+[\kappa_q] = y_p[\kappa_q]$ (synchronized output).

The synchronization requirements could be easily fulfilled if the sequences of switching times, of parameter values, and of state jumps were defined *a priori* and available to both $\mathcal{W}$ and $\mathcal{Q}$. Also, the switch times and jump synchronization requirements alone could be trivially met if the watermark generator and remover had a second channel of communication, for the sole purpose of exchanging the switch and jump information. However, both these solutions would greatly reduce the applicability and the inherent robustness against adversaries.

Instead, we propose a solution which we name *implicit synchronization*, where the triggering is decided by the generator $\mathcal{W}$ and no additional data exchange with $\mathcal{Q}$ is needed apart from the existing communication of the watermarked data y_w. Moreover, it is also desired that the implicit synchronization protocol has reduced visibility to the adversary, as to decrease the leakage of information about the filter parameter changes to the adversary. A first simple protocol is provided as an example below, while the next section details a more general switching protocol design.

C. Illustrative Example

The switching protocol will be presented along the synchronization requirements outlined above.

1) Switch Time and Output Synchronization: Let $\mathbb{1}\{x > a\}$ denote the indicator function of the condition $x > a$, where $\mathbb{1}\{x > a\} = 1$ if $x > a$, and 0 otherwise. The triggering function τ_q at $\mathcal{Q}$ is defined as $\tau_q[k] = \mathbb{1}\{|y_q^-[k] - y_p[k-1]| > \delta^*\}$, where δ^* is a design parameter, $y_q^-[k] = C_q(\theta_q^-)x_q^-[k] + D_q(\theta_q^-)y_w[k]$, and $y_w[k]$ is the data received from the watermark generator.

As for the triggering function τ_w at $\mathcal{W}$, it is constructed as $\tau_w[k] = \mathbb{1}\{\tau'[k] = 1 \vee \hat{\tau}_q[k] = 1\}$, which has a controlled component that can be arbitrarily decided, denoted as $\tau'[k]$, and a noncontrolled part that predicts a spontaneous switch at $\mathcal{Q}$, defined as

$$\begin{aligned} \hat{\tau}_q[k] &= \mathbb{1}\left\{|y_q^-[k] - y_p[k-1]| > \delta^*\right\} \\ &= \mathbb{1}\left\{| - D_w^{-1}C_w(\theta_w^-)x_w^-[k] + D_w^{-1}(\theta_w^-)y_w[k] \right. \\ &\qquad \left. -y_p[k-1]| > \delta^*\right\}. \end{aligned} \tag{10}$$

To ensure switch time synchronization, whenever $\tau_w[\kappa_w] = 1$, $\mathcal{W}$ modifies its transmitted data from $y_w^-[\kappa_w]$ to $y_w^+[\kappa_w]$, where $y_w^+[\kappa_w]$ is constructed so that it induces a switch at $\mathcal{Q}$. For instance, given the triggering function τ_q defined earlier, the data $y_w^+[\kappa_w]$ may be computed as

$$\begin{aligned} y_w^+[\kappa_w] = \arg\min_{y\in\mathbb{R}} \quad & |y - y_w^-[\kappa_w]| \\ \text{s.t.} \quad & |y_q^-[\kappa_w] - y_p(\kappa_w - 1)| > \bar{\delta}^\star \end{aligned} \tag{11}$$

with $\bar{\delta}^\star = \delta^\star + \dfrac{|y_p[\kappa_w] - y_p[\kappa_w - 1]|^2}{1 + |y_p[\kappa_w] - y_p[\kappa_w - 1]|}$.

As for output synchronization, by replacing $\delta^\star$ with a suitable function $\bar{\delta}^\star$ in the switching condition, we ensure that $y_p[\kappa_w]$ can be uniquely retrieved at $\mathcal{Q}$ from its local information and the

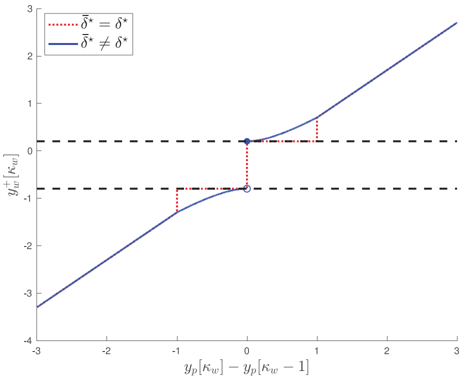

Fig. 2. Plot of the watermarked output $y_w^+[\kappa_w]$ at switching times, as a function of $y_p[\kappa_w] - y_p[\kappa_w - 1]$, for the exact switching rule (dotted line, $\bar{\delta}^\star = \delta^\star$) and for the modified switching rule (solid line, $\bar{\delta}^\star \neq \delta^\star$). The area between the dashed lines represents the region where no switch would be triggered at $\mathcal{Q}$.

received watermarked measurement $y_w^+[\kappa_w]$. As an illustration, Fig. 2 shows that the mapping $f(\cdot)$ from $y_p[\kappa_w] - y_p[\kappa_w - 1]$ to $y_w^+[\kappa_w]$, defined by (11), is invertible over the domain of $y_w^+[\kappa_w]$. On the other hand, when $\bar{\delta}^\star = \delta^\star$, this mapping is not invertible in the entire domain.

By using the proposed scheme, the original measurement $y_p[\kappa_w]$ can be retrieved at the remover $\mathcal{Q}$ as $y_q^+[\kappa_w] = f^{-1}(y_w^+[\kappa_w]) + y_p[\kappa_w - 1]$.

2) Switch and Jump Synchronization: Once switch time synchronization is ensured, keeping the filters matched amounts to selecting a shared sequence of filter parameters. Therefore, we design the switch functions σ_w and σ_q *a priori* to be identical and ensure that each parameter of the sequence guarantees stability of $\mathcal{W}$ and $\mathcal{Q}$.

Finally, to ensure the states of the filters are synchronized at switching times, suitable jump rules $\rho_w(\cdot)$ and $\rho_q(\cdot)$ should be designed. For instance, at switching times, $\rho_w(\cdot)$ can be designed as

$$\begin{aligned}\rho_w(\cdot) &= \arg\min_{x_w} \quad \|x_w\|_2^2\\ &\qquad \text{s.t.} \quad y_w^+[\kappa_w] = C_w(\theta_w^+)x_w + D_w(\theta_w^+)y_p[\kappa_w].\end{aligned}$$

We highlight that the jump function ρ_w is in fact a composite function, where one first computes the jump in the watermarked output, $y_w^+[\kappa_w]$, based on which a consistent jump in the state is computed, $x_w^+[\kappa_w]$.

Similarly, at $\mathcal{Q}$ the state jump function is constructed as

$$\begin{aligned}\rho_q(\cdot) &= \arg\min_{x_q} \quad \|x_q\|_2^2\\ &\qquad \text{s.t.} \quad D_q(\theta_q^+)y_w^+[\kappa_q] = -C_q(\theta_q^+)x_q + y_q^+[\kappa_q].\end{aligned}$$

As long as $\mathcal{W}$ and $\mathcal{Q}$ are switch synchronized and matched, then we can straightforwardly verify that the jump policies yield $x_w^+[\kappa_w] = x_q^+[\kappa_q]$.

In Section IV, a generic protocol design is detailed, with definitions and characterization of key properties that ensure the feasibility and correct behavior of the protocol.

IV. Low Visibility Switching Protocol

This section describes the general characteristics of event-triggered watermarking switching protocols that enable the synchronous update of the filters' parameters and initial conditions at the generator and remover. Before the design, several supporting concepts are first defined.

A. Defining Synchronization

We begin by defining the building blocks related to synchronization, which also include the triggering functions in Definition 3.

Definition 4: The generator $\mathcal{W}$ and remover $\mathcal{Q}$ are said to be *synchronized* at switching time $k \in \mathbb{N}$ if they are

1) *Trigger-synchronized, i.e.*, $\tau_w(y_p[k]) = \tau_q(y_w[k])$;
2) *Switch-synchronized, i.e.*, $\theta_w^+[k] = \theta_q^+[k]$;
3) *Jump-synchronized, i.e.*, $x_w^+[k] = x_q^+[k]$;
4) *Output-synchronized, i.e.*, $y_p[k] = y_q^+[k]$.

Essentially, the main objective of the switching protocol is to ensure that Definition 4 holds. However, to be implementable, the protocol must comply with the information structures available at $\mathcal{W}$ and $\mathcal{Q}$, respectively.

B. Defining Local Information and Implicit Synchronization

In the following, different information sets that constrain the protocol implementation are defined. The sets are defined in terms of input and state trajectories over a time interval of size $N_\mathcal{I} \geq 1$, since the last switching time instant, with $N_\mathcal{I} \in \mathbb{N}$ being a design parameter.

Definition 5 (Information at $\mathcal{W}$): The set $\mathcal{I}_w[k] \triangleq \{Y_{p,(k-N_\mathcal{I},k]}, x_w[k - N_\mathcal{I}]\}$ is the *local information available at* $\mathcal{W}$ at time instant k.

Definition 6 (Information at $\mathcal{Q}$): The set $\mathcal{I}_q[k] \triangleq \{Y_{w,(k-N_\mathcal{I},k]}, x_q[k - N_\mathcal{I}]\}$ and $\mathcal{I}_q^+[\kappa_w] \triangleq \mathcal{I}_q[\kappa_w - 1] \cup \{y_w^+[\kappa_w]\}$ are the *local information available on* $\mathcal{Q}$ at time instant k and after a switch at time $\kappa_w \in \mathcal{K}_w$, respectively.

As discussed in the previous example, the switching protocol relies on $\mathcal{W}$ tracking the spontaneous switches at $\mathcal{Q}$, and inducing a forced switch on $\mathcal{Q}$ at switching times $\kappa_w \in \mathcal{K}_w$ by replacing $y^-[\kappa_w]$ with $y^+[\kappa_w]$. Therefore, the set $\mathcal{I}_q^+[\kappa_w]$ plays a central role in the switching protocol. A few additional remarks are in order, to highlight the relations between the above information sets.

Remark 2: Under the assumption that the watermarking filters are initially synchronized, recall from Theorem 1 that $x_w[k] = x_q[k]$ holds in between switching times. Hence, since $y_w[k]$ is computed based on $\mathcal{I}_w[k]$, one can directly conclude that $\mathcal{I}_q[k] \subset \mathcal{I}_w[k]$ holds in between switching times.

Remark 3: Under the assumption of synchronous switching, *i.e.*, $\mathcal{K}_w = \mathcal{K}_q$, at switching times $\kappa_w \in \mathcal{K}_w$ the relation $\mathcal{I}_q^+[\kappa_w] \subset \mathcal{I}_w[\kappa_w]$ holds, since the data $y_w^+[\kappa_w]$ is computed based on $\mathcal{I}_w[\kappa_w]$.

Given the above remarks, we observe that the information available at the watermark remover $\mathcal{Q}$ is also available at the watermark generator $\mathcal{W}$. This observation is the basis for achieving implicit synchronization between these filters, which is defined as follows.

Definition 7 (Implicit Synchronization): A pair $(\mathcal{W},\ \mathcal{Q})$ of, respectively, a switching watermark generator and remover is said to be *implicitly synchronized* if at time k

1) the triggering sets $\mathcal{C}_w$ and $\mathcal{C}_q$ are parameterized, respectively, by $\mathcal{I}_w[k-1]$ and $\mathcal{I}_q[k-1]$, which is denoted as $\mathcal{C}_w(\mathcal{I}_w[k-1])$ and $\mathcal{C}_q(\mathcal{I}_q[k-1])$;
2) the state jump functions ρ_w and ρ_q are parameterized by $\mathcal{I}_w[\kappa_w]$ and $\mathcal{I}_q^+[\kappa_q]$, respectively;
3) the pair $(\mathcal{W},\ \mathcal{Q})$ is synchronized.

C. Defining Switch Visibility

In addition to ensuring synchronization, it is also desirable to prevent an eavesdropping MITM attacker to detect switching instants. A switching protocol may be evaluated with respect to such an objective by means of a switch visibility metric that penalizes deviations between $y_w^+[\kappa_w]$ and $y_w^-[\kappa_w]$. A generic metric may be defined as follows.

Definition 8: A function $g(y_w^+[\kappa_w],\ y_p[\kappa_w] | \mathcal{I}_q[\kappa_w-1]) : \mathbb{R}^{n_y} \times \mathbb{R}^{n_y} \mapsto \mathbb{R}$ is said to be a *switch visibility metric* if, for any y_p, it is convex on y_w^+, even on y_w^+ around its minimum, and bounded from below. Furthermore, its global unconstrained minimizer with respect to $y_w^+[\kappa_w]$ is denoted by $\alpha(y_p[\kappa_w]\,|\,\mathcal{I}_q[\kappa_w-1]) \triangleq \arg\min_y g(y, y_p[\kappa_w]\,|\,\mathcal{I}_q[\kappa_w-1])$.

D. Switching Protocol Design

As described earlier in Section III-B, our proposed switching protocol has two nontrivial stages: first ensure *switch time and output synchronization*, and then agree on switch and state jumps that maintain *switch and jump synchronization*.

Switch time synchronization involves that $\mathcal{W}$ tracks possible spontaneous switches at $\mathcal{Q}$, and then chooses a suitable $y_w^+[\kappa_w]$ that induces a switch on $\mathcal{Q}$ under the constraints of implicit synchronization, while ensuring output synchronization. Switch synchronization is trivially achieved once switch time synchronization is ensured. Finally, jump synchronization requires that, at switch times, $\mathcal{W}$ and $\mathcal{Q}$ agree on state jumps $x_w^+[\kappa_w] = x_q^+[\kappa_w]$ that are consistent with $y_w^+[\kappa_w]$. The remainder of this section discusses in detail these two stages.

1) Switch Time and Output Synchronization:

Recall that the switching functions at the generator $\mathcal{W}$ and at the remover $\mathcal{Q}$ are defined in terms of the corresponding triggering sets $\mathcal{C}_w$ and $\mathcal{C}_q$, respectively

$$\begin{aligned}\tau_w(y_p[k]) &= \mathbb{1}\left\{y_p[k] \notin \mathcal{C}_w\left(\mathcal{I}_w[k-1]\right)\right\}\\ \tau_q(y_w[k]) &= \mathbb{1}\left\{y_w[k] \notin \mathcal{C}_q\left(\mathcal{I}_q[k-1]\right)\right\}.\end{aligned} \tag{12}$$

The watermark generator $\mathcal{W}$ must be able to force arbitrary switches when requested, as well as to track spontaneous switches triggered at $\mathcal{Q}$. Thus, the first step of the protocol design is to define the triggering set $\mathcal{C}_w$ as

$$\mathcal{C}_w \triangleq \begin{cases}\hat{\mathcal{C}}_q\left(\mathcal{I}_w[k-1]\right) & \text{, if } \tau_w'[k]=0\\ \emptyset & \text{, if } \tau_w'[k]=1\end{cases} \tag{13}$$

where $\hat{\mathcal{C}}_q(\mathcal{I}_w[k-1])$ is defined as

$$\begin{aligned}\hat{\mathcal{C}}_q &\triangleq \left\{y_p[k]:\ y_w^-[k] \in \mathcal{C}_q\left(\mathcal{I}_w[k-1]\right)\right\}\\ &= \left\{y:\ C_w x_w[k] + D_w y \in \mathcal{C}_q\left(\mathcal{I}_q[k-1]\right)\right\}.\end{aligned} \tag{14}$$

The switch-forcing function $\tau_w'[k]$ can be defined to ensure, for instance, that switches occur often enough independently of the characteristics of the signal $y_p[k]$, thus helping the detection of replay attacks as analyzed in Section VI.

Having defined the triggering sets, we can now characterize the proposed low-visibility switching protocol in terms of choosing a suitable $y_w^+[k]$ according to the following requirements, which should hold at each $\kappa_w \in \mathcal{K}_w$:

R1. The visibility of the switch should be reduced, that is, $g(y_w^+[\kappa_w],\ y_p[\kappa_w] | \mathcal{I}_q[\kappa_w-1])$ should be minimized.

R2. $y_w^+[\kappa_w] \triangleq f(y_p[\kappa_w] | \mathcal{I}_q[\kappa_w-1])$ should trigger a parameter switch at the remover's side, i.e., it should satisfy the triggering condition $y_w^+[\kappa_w] \notin \mathcal{C}_q(\mathcal{I}_q[\kappa_w-1])$.

R3. The scheme should allow for the remover to compute $y_p[\kappa_w]$, based on its available information $\mathcal{I}_q^+[\kappa_w]$. In other words, $f(\cdot|\mathcal{I}_q[\kappa_w-1])$ must be an invertible function of the newly received data $y_w^+[\kappa_w]$ and $\mathcal{I}_q[\kappa_w-1]$, which together correspond to $\mathcal{I}_q^+[\kappa_w]$.

The three requirements just introduced naturally lead to implementing the function f as the solution to a constrained optimization problem. In particular, assuming $\mathcal{W}$ and $\mathcal{Q}$ were synchronized at time $\kappa_w - 1$, a function f satisfying requirements R1 and R2 above can be formulated as

$$\begin{aligned}y_w^+[\kappa_w] = \arg\min_{y\in\mathbb{R}} \quad & g\left(y,\ y_p[\kappa_w] | \mathcal{I}_q[\kappa_w-1]\right)\\ \text{s.t.} \quad & y \notin \mathcal{C}_q\left(\mathcal{I}_q[\kappa_w-1]\right)\end{aligned} \tag{15}$$

However, this formulation does not comply with the third requirement of the protocol.

Lemma 2: The function $y_w^+[\kappa_w] = f(y_p[\kappa_w] | \mathcal{I}_q[\kappa_w-1])$ defined as (15) is not invertible.

Proof: The proof follows directly from the fact that, for all values of y_p, where the global minimizer of g lies inside the set $\mathcal{C}_q$, the optimization problem (15) corresponds to a projection (with respect to g) of the minimizer onto the boundary of $\mathcal{C}_q$. Hence, values of y_p along the same projection direction will lead to the same optimal solution, and thus f is not invertible. ■

The above result illustrates how the event-triggering constraint may affect the invertibility of $f(\cdot)$. We must, therefore, consider a modified constraint that also depends on $y_p[\kappa_w]$, while complying with requirement R2 of allowing the remover to detect the switching event. We shall first consider an equivalent formulation of the constraint $y \notin \mathcal{C}_q(\mathcal{I}_q[\kappa_w-1])$.

Lemma 3: The constraint $y \notin \mathcal{C}_q(\mathcal{I}_q[\kappa_w-1])$ on the real variable $y \in \mathbb{R}$ can be rewritten as $|y - \beta(\mathcal{I}_q[\kappa_w-1])| \geq \delta(\mathcal{I}_q[\kappa_w-1])$, for some real-valued functions $\beta(\mathcal{I}_q[\kappa_w-1])$ and $\delta(\mathcal{I}_q[\kappa_w-1]) > 0$.

Proof: The proof immediately follows from the fact that $\mathcal{C}_q(\mathcal{I}_q[\kappa_w - 1])$ is a convex set on the real line, which means that it can be defined as $\mathcal{C}_q \triangleq \{y \in \mathbb{R} : \ |y - \beta| < \delta\}$ for some $\beta \in \mathbb{R}$ and $\delta > 0$. ■

Given the above formulation, we define the set $\overline{\mathcal{C}}(y_p[\kappa_w]| \, \mathcal{I}_q[\kappa_w - 1]) \triangleq \{y \in \mathbb{R} : \quad |y - \beta(\mathcal{I}_q[\kappa_w - 1])| < \delta(\mathcal{I}_q[\kappa_w - 1]) + \delta_p(y_p[\kappa_w]| \, \mathcal{I}_q[\kappa_w - 1])$, where $\delta_p(y_p[\kappa_w]| \, \mathcal{I}_q [\kappa_w - 1])$ is a real valued, invertible, positive function of $y_p[\kappa_w]$, parameterized by the information available at the remover. Accordingly, we shall replace the triggering constraint $y \notin \mathcal{C}_q(\mathcal{I}_q[\kappa_w])$ with $y \notin \overline{\mathcal{C}}(y_p[\kappa_w]|\mathcal{I}_q[\kappa_w])$, and consider instead the following modified problem to define the function $f(\cdot| \, \mathcal{I}_q[\kappa_w - 1])$:

$$\begin{aligned} y_w^+[\kappa_w] = \arg\min_{y \in \mathbb{R}} \quad & g\,(y, \ y_p[\kappa_w]| \, \mathcal{I}_q[\kappa_w - 1]) \\ \text{s.t.} \quad & |y - \beta(\mathcal{I}_q[\kappa_w - 1])| \geq \delta(\mathcal{I}_q[\kappa_w - 1]) \\ & \qquad + \delta_p\,(y_p[\kappa_w]| \, \mathcal{I}_q[\kappa_w - 1])\,. \end{aligned} \tag{16}$$

Note that the constraint in (16) is now a function of $y_p[\kappa_w]$, while it still ensures that the switching condition for $\mathcal{Q}$ is satisfied, since $\mathcal{C}_q(\mathcal{I}_q[\kappa_w - 1]) \subset \overline{\mathcal{C}}(y_p[\kappa_w]| \, \mathcal{I}_q[\kappa_w - 1])$. In the following, we often drop the argument $\mathcal{I}_q[\kappa_w - 1]$ when there is no risk of ambiguity.

As stated earlier, the aim is to design the optimization problem (16), namely to design the function $g(y, \ y_p[\kappa_w])$ and the variables β, and $\delta_p(y_p[\kappa_w])$, such that the requirements of the switching scheme are met. Clearly, the proposed optimization problem satisfies requirements R1 and R2, while requirement R3 shall be discussed in the following.

In order to analyze the last requirement R3, we must derive the optimal solution to (16).

Lemma 4: Given the optimization problem (16) and Definition 8, define the functions $\Delta g(x, z) \triangleq g(x, \ y_p[\kappa_w]) - g(z, \ y_p[\kappa_w])$, $\quad y_1(y_p[\kappa_w]) \triangleq \beta + \delta + \delta_p(y_p[\kappa_w])$, and $y_2(y_p[\kappa_w]) \triangleq \beta - (\delta + \delta_p(y_p[\kappa_w]))$. The optimal solution to (16) is given by

$$\begin{cases} \alpha(y_p[\kappa_w]), & \text{if } \alpha(y_p[\kappa_w]) \notin \overline{\mathcal{C}} \\ y_1\,(y_p[\kappa_w])\,, & \text{if } \alpha(y_p[\kappa_w]) \in \overline{\mathcal{C}} \text{ and } \Delta g(y_1, y_2) \leq 0 \\ y_2\,(y_p[\kappa_w])\,, & \text{if } \alpha(y_p[\kappa_w]) \in \overline{\mathcal{C}} \text{ and } \Delta g(y_1, y_2) > 0. \end{cases} \tag{17}$$

Proof: Recalling the properties of g, the proof follows from observing that (16) is a projection of the global unconstrained minimizer of g onto the constraint set, namely the complement of $\overline{\mathcal{C}}$. Since $\overline{\mathcal{C}}$ is a convex interval on the real line, the optimal solution candidates are either the global minimizer of g $(\alpha(y_p[\kappa_w]))$, or the two extremes of $\overline{\mathcal{C}}$ (y_1 and y_2). ■

The next result immediately follows.

Lemma 5: The function $y_w^+[\kappa_w] = f(y_p[\kappa_w]| \, \mathcal{I}_q[\kappa_w - 1])$ defined as (16) is not invertible, if $\alpha(y_p[\kappa_w]| \, \mathcal{I}_q[\kappa_w - 1])$ is not invertible.

Proof: The proof follows from the characterization of the optimal solution (17). ■

Algorithm 1: Switching Protocol Ensuring Switch Time and Output Synchronization.

```
 At the generator $\mathcal{W}$:
1:  while $y_p[k] \in \mathcal{C}_w(\mathcal{I}_w[k-1])$ do
2:      wait for next time instant $k$
3:  end while
4:  $\kappa_w \leftarrow k$ we have a switch
5:  $y_w^+[\kappa_w] \leftarrow$ solution of (16)
6:  goto 1.
At the remover $\mathcal{Q}$:
1:  while $y_w[k] \in \mathcal{C}_q(\mathcal{I}_q[k-1])$do
2:      wait for next time instant $k$
3:  end while
4:  $\kappa_q \leftarrow k$ we have a switch
5:  $y_q^+[\kappa_q] \leftarrow \alpha^{-1}(\hat{\alpha}(y_w^+[k_a])| \, \mathcal{I}_q[\kappa_q - 1])$ from (18)
6:  goto 1.
```

Although the previous result points to a necessary property for $f(\cdot)$ to be invertible, is it not clear whether these conditions are also sufficient. Next, we propose a slightly more restrictive definition of the variable $\delta_p(y_p[\kappa_w])$ that ensures sufficiency.

Lemma 6:

Let $\delta_p(y_p[\kappa_w])$ be a positive, monotonically increasing, and invertible function of $|\alpha(y_p[\kappa_w]) - \beta|$, and recall that, by definition, $g(y, \ y_p[\kappa_w])$ is an even function with respect to its unconstrained global minimizer $\alpha(y_p[\kappa_w])$. Then, given $\mathcal{I}_q^+[\kappa_w]$, the value of the global minimizer $\alpha(y_p[\kappa_w])$ can be retrieved by $\mathcal{Q}$ as (18).

Proof: The proof may be found in the appendix. ■

Theorem 2: The function $y_w^+[\kappa_w] = f(y_p[\kappa_w])$ defined as (16) is invertible, if $\alpha(y_p[\kappa_w])$ is invertible, $g(y, \ y_p[\kappa_w])$ is an even function with respect to $\alpha(y_p[\kappa_w])$, and $\delta_p(y_p[\kappa_w])$ is a positive, monotonically increasing, and invertible function of $|\alpha(y_p[\kappa_w]) - \beta|$. Furthermore, the plant output at switching time κ_w, $y_p[\kappa_w]$, can be reconstructed at $\mathcal{Q}$ as $y_p[\kappa_w] = y_q^+[\kappa_w] \triangleq \alpha^{-1}(\hat{\alpha}(y_w^+[\kappa_w]))$.

Proof: The proof follows from Lemma 6, which determines that the value of $\alpha(y_p[\kappa_w])$ can be obtained by $\mathcal{Q}$ as $\hat{\alpha}(y_w^+[\kappa_w])$ in (18). Finally, since $\alpha(y_p[\kappa_w])$ is an invertible function, the original plant measurement can be reconstructed as $y_p[\kappa_w] = \alpha^{-1}(\hat{\alpha}(y_w^+[\kappa_w]))$, which concludes the proof. ■

$$\hat{\alpha}(y_w^+[\kappa_w]) = \begin{cases} y_w^+[\kappa_w], \text{if } |y_w^+[\kappa_w] - \beta| \leq \delta_p^{-1}\,(|y_w^+[\kappa_w] - \beta| - \delta) \\ \beta + \text{sign}\,(y_w^+[\kappa_w] - \beta)\,\delta_p^{-1}\,(|y_w^+[\kappa_w] - \beta| - \delta)\,, \quad \text{otherwise.} \end{cases} \tag{18}$$

Combining the formulations proposed in this section, the switching protocol and the recovery of $y_p[\kappa_w]$ by the remover $\mathcal{Q}$ can be summarized in Algorithm 1.

Now that the triggering of $\mathcal{Q}$ and its synchronization to $\mathcal{W}$ have been addressed, we will shift our attention to the last components of the synchronization protocol that must be defined: the switch maps σ_w and σ_q and the jump maps ρ_w and ρ_q.

2) Switch and Jump Synchronization: Having designed the switching protocol to achieve switch synchronization, we now address the second stage of the protocol: ensuring switch and jump synchronization.

Theorem 2 summarizes the computation of $y_w^+[\kappa_w]$ that triggers a switch at the remover at $\kappa_q = \kappa_w$ and enables it to construct the value of $y_q^+[\kappa_q] = y_p[\kappa_w]$. Switch synchronization is trivially achieved once switch time synchronization is ensured, by designing the switch maps σ_w and σ_q as autonomous sequences.

The remaining task is to define the jump functions $\rho_w(\mathcal{I}_w[\kappa_w])$ and $\rho_q(\mathcal{I}_q^+[\kappa_q])$ producing consistent state jumps $x_w^+[\kappa_w]$ and $x_q^+[\kappa_q]$ satisfying the following relations:

$$y_w^+[\kappa_w] = C_w(\theta_w^+)x_w^+[\kappa_w] + D_w(\theta_w^+)y_p[\kappa_w]$$

$$y_q^+[\kappa_q] = C_q(\theta_q^+)x_q^+[\kappa_q] + D_q(\theta_q^+)y_w^+[\kappa_q].$$

Note that these equations are equivalent, if $\mathcal{W}$ and $\mathcal{Q}$ are matched and switch synchronized, given the relations in Lemma 1 and $y_q^+[\kappa_q] = y_p[\kappa_w]$. Hence, we next describe the function $\rho_w(\mathcal{I}_w[\kappa_w])$, and let $\rho_q(\mathcal{I}_q^+[\kappa_q]) = \rho_w(\mathcal{I}_w[\kappa_w])$.

Since $C_w(\theta_w^+) \in \mathbb{R}^{1\times N}$, there may exist multiple solutions to $x_w^+[\kappa_w]$. To address this, we define a strongly convex function $h(x_w)$ and obtain $x_w^+[\kappa_w] = \rho_w(\mathcal{I}_w[\kappa_w])$ as

$$\begin{aligned} x_w^+[\kappa_w] = \arg\min_{x_w} \quad & h(x_w) \\ \text{s.t.} \quad & C_w(\theta_w^+)x_w = y_w^+[\kappa_w] - D_w(\theta_w^+)y_p[\kappa_w]. \end{aligned} \tag{19}$$

Remark 4: Although we assumed in (1) the presence of physical modeling and measurement uncertainties, we implicitly assumed that watermarked data is transmitted over a noiseless, lossless digital network. Such ideal condition allowed us to prove that $\mathcal{W}$ and $\mathcal{Q}$ remain implicitly synchronized and the closed loop performances are not modified by the watermarking. The only "uncertainty" that could cause loss of synchronicity is indeed the attacker presence, and how this would ease attack detection will be discussed in Section VI.

V. Stability Analysis

In earlier sections, we have presented the watermarking generator and remover as hybrid discrete-time systems, and designed the scheme as to ensure nominal performance and parameter switching without additional communication costs. However, stability of the proposed scheme has not been addressed yet.

In this section, we report first results regarding the stability of the closed-loop system with the proposed watermarking scheme in two cases: synchronized filters and nonsynchronized filters over interswitching intervals (i.e., with constant mismatched parameters).

A. Synchronized Filters

The case of synchronized filters is considered first, for which the plant output is decoupled from the filters' states.

Theorem 3: Let the generator $\mathcal{W}$ and the remover $\mathcal{Q}$ be synchronized at all times. Then the closed-loop system is asymptotically stable, i.e., $x_p[k]$, $x_{cr}[k]$, and $y_p[k]$ converge asymptotically to the origin. Moreover, if $h(x) = \|x\|$, the internal states of the generator and remover, $x_w[k]$ and $x_q[k]$, are uniformly ultimately bounded.

Proof: The proof may be found in the appendix. ■

B. Nonsynchronized Filters Over Interswitching Intervals

Determining stability of the closed-loop system with nonsynchronized filters and mismatched parameters is a robust stability problem with multiplicative model uncertainty, where the uncertainty is in fact a hybrid system.

In the following, we restrict our attention to the interswitching times, during which the uncertainty behaves as a linear time-invariant system. We start by formulating the nominal system and the uncertainty under analysis.

The key steps are to rewrite $\tilde{y}_p[k] = y_q[k]$ as $\tilde{y}_p[k] = y_p[k] + \Delta y_q[k]$, where $\Delta y_q[k]$ is the output of the system $\mathcal{D}(\theta_w, \theta_q)$ described by (9), and to consider the nominal closed-loop system from the input $\Delta y_q[k]$ to the output $y_p[k]$, namely $\mathcal{S}_{\Delta y_q, y_p}$ given by

$$\begin{aligned} \begin{bmatrix} x_p[k+1] \\ x_{cr}[k+1] \end{bmatrix} &= \begin{bmatrix} A_p + B_pD_uC_p & B_pC_u \\ B_{cr}C_p & A_{cr}[k] \end{bmatrix} \begin{bmatrix} x_p[k] \\ x_{cr}[k] \end{bmatrix} \\ &\quad + \begin{bmatrix} B_pD_u \\ B_{cr} \end{bmatrix} \Delta y_q[k] \\ y_p[k] &= \begin{bmatrix} C_p & 0 \end{bmatrix} \begin{bmatrix} x_p[k] \\ x_{cr}[k] \end{bmatrix}. \end{aligned} \tag{20}$$

Then, the perturbed closed-loop system can be described as the nominal closed-loop system, $\mathcal{S}_{\Delta y_q, y_p}$, interconnected with $D(\theta_w, \theta_q)$. Defining $\gamma(\Sigma)$ as the $\mathcal{H}_\infty$-norm of a linear system Σ, the following stability result directly follows.

Theorem 4: Let the generator $\mathcal{W}$ and the remover $\mathcal{Q}$ be nonsynchronized at a switching time instant κ_i, and assume no future switching occurs. Then, the closed-loop system and watermarking filters are robustly asymptotically stable if $\gamma(\mathcal{S}_{\Delta y_q, y_p})\gamma(\mathcal{D}(\theta_w[\kappa_i], \theta_q[\kappa_i])) \leq 1$.

Proof: The proof follows from classical results on robust stability (see for instance [22]). ■

Although Theorem 4 gives only a sufficient condition, it allows for a simpler design of the filter parameters, by imposing two $\mathcal{H}_\infty$-norm constraints for each pair of filter parameters. The next results formalize this statement.

Corollary 1: Let the generator $\mathcal{W}$ and the remover $\mathcal{Q}$ be nonsynchronized at a switching time instant k_i, and assume no future switching occurs. Then, the closed-loop system and watermarking filters are robustly asymptotically stable if $\mathcal{W}(z;\, \theta_i)$, $\mathcal{W}^{-1}(z;\, \theta_i)$, $\mathcal{W}(z;\, \theta_j)$, and $\mathcal{W}^{-1}(z;\, \theta_j)$ are stable for all choice of filter parameters $\theta_i, \theta_j \in \Theta$, and, for all $\theta_i, \theta_j \in \Theta$, $\theta_j \neq \theta_i$, the following frequency domain constraints are satisfied for all $z \in \mathbb{C}$ on the unit circle

$$|\left(\mathcal{W}(z;\theta_i) - \mathcal{W}(z;\theta_j)\right)| \leq \gamma\left(\mathcal{S}_{\Delta y_q, y_p}\right)^{-1} |\mathcal{W}(z;\theta_j)|. \tag{21}$$

Proof: The proof follows directly from Theorem 4. First note that $\gamma(\mathcal{D}(\theta_i, \theta_j))$ is finite if and only if both generator filters $W(z;\, \theta_i)$ and $W(z;\, \theta_j)$ and their inverses are stable.

The inequalities follow by recalling that $\gamma(\mathcal{D}(\theta_i,\theta_j)) = \sup_{|z|=1} |\mathcal{D}(z;\,\theta_i,\theta_j)|$, and $\mathcal{D}(z;\,\theta_i,\theta_j) = \mathcal{W}(z;\,\theta_i)\mathcal{W}^{-1}(z;\,\theta_j) - 1$, from which we derive $\gamma(\mathcal{D}(\theta_i,\theta_j)) = \sup_{|z|=1} |\mathcal{W}(z;\,\theta_i) - \mathcal{W}(z;\,\theta_j)||\mathcal{W}^{-1}(z;\,\theta_j)|$.

Thus, we conclude that $\gamma(\mathcal{S}_{\Delta y_q, y_p})\gamma(\mathcal{D}(\theta_i,\theta_j)) \leq 1$ is equivalent to the inequality (21) for all possible combinations of θ_i and θ_j. ■

Note that these frequency domain inequalities ensuring robust stability could be enforced by requiring different parameters θ_i and θ_j to be sufficiently close, depending on the $\mathcal{H}_\infty$-norm of the nominal closed-loop system. On the other hand, to enable the detection of the mismatch and replay attacks, one desires that the filter parameters are as different as possible. Therefore, one must tradeoff robust stability and detectability of filter mismatches.

VI. Detection of MITM Attacks

In this section, we address the detection of MITM attacks. We will design and analyze here the anomaly detector $\mathcal{R}$ depicted in Fig. 1 and whose dynamics has been introduced in (2). By leveraging the approach introduced in [16]–[18], we will build it around the following estimator:

$$\hat{\mathcal{P}}: \begin{cases} \hat{x}_p[k+1] = A_p\hat{x}_p[k] + B_p u[k] + K\left(y_q[k] - \hat{y}_p[k]\right) \\ \hat{y}_p[k] = C_p\hat{x}_p[k] \end{cases} \tag{22}$$

where $\hat{x}_p \in \mathbb{R}^{n_p}$ and $\hat{y}_p \in \mathbb{R}^{n_y}$ are, respectively, dynamic estimates of the plant vectors x_p and y_p. Before proceeding further, we need also to recall here the following assumptions, for the sake of well-posedness.

Assumption 4: No attacks are present for $0 \leq k < k_a$, with k_a being the attack start time.

Assumption 5: (A_p, C_p) is a detectable pair.

The observer gain K is chosen such that $A_r \triangleq A_p - KC_p$ is a Schur matrix. Such a choice is always possible, thanks to Assumption 5. The dynamics of $\mathcal{R}$ can be obtained from the ones of $\hat{\mathcal{P}}$ by defining the output residual as $y_r \triangleq y_q - \hat{y}_p$ and by setting $x_r = \hat{x}_p$, $A_r = A_p - KC_p$, $B_r = B_p$, $K_r = K$, $C_r = -C_p$, $D_r = 0$, $E_r = I_{n_y}$.

When no attack is present, the dynamics of the estimation error $\epsilon \triangleq x_p - \hat{x}_p$ and the detection residual y_r can thus be written by subtracting (22) from $\mathcal{P}$ dynamics in (1), obtaining

$$\begin{cases} \epsilon[k+1] = A_r\epsilon[k] - K\xi[k] + \eta[k] \\ y_r[k] = C_p\epsilon[k] + \xi[k]. \end{cases} \tag{23}$$

Recalling the definition of the detection threshold $\bar{y}_r$ in (3), we can write the dynamical solution for its ith component, $i \in \{1, \ldots, n_y\}$, as

$$\bar{y}_{r,(i)}[k] \triangleq \nu_1^i \left[\sum_{h=0}^{k-1} \left(\nu_2^i\right)^{k-1-h} \left(\bar{\eta}[h] + \|K\|\bar{\xi}[h]\right) + \left(\beta^i\right)^k \bar{\epsilon}[0]\right] + \bar{\xi}[k] \tag{24}$$

following known results from [17] and [23]. The two constants ν_1^i and ν_2^i are such that $\|C_{p,(i)}(A_r)^k\| \leq \nu_1^i(\nu_2^i)^k \leq \|C_{p,(i)}\| \cdot \|(A_r)^k\|$ with $C_{p,(i)}$ being the ith row of matrix C_p. Furthermore, $\bar{\eta}$, $\bar{\epsilon}[0]$, and $\bar{\xi}$ are upper bounds on the norms of, respectively, η, $\epsilon[0]$, and ξ.

Lemma 7: The adaptive detection threshold (24) will not lead to false alarms, that is $|y_{r,(i)}[k]| \leq \bar{y}_{r,(i)}[k]$ for all $1 \leq i \leq n_y$ and $0 \leq k < k_a$.

Proof: It follows by definition of the threshold in (24), from Assumption 1 and from analogous results in [17] and [23]. ■

A. Effect of Attacks

When the proposed watermarking approach is in place, an MITM attacker would no longer be able to directly affect the plant output y_p as shown in (4), but will instead affect the watermarked output according to $\tilde{y}_w[k] = \phi(Y_{w,(k-\tilde{N},k]})$. For easing the subsequent analysis, we will equivalently write such effect as an additive term

$$\tilde{y}_w[k] = y_w[k] + \varphi[k] \tag{25}$$

where it simply holds $\varphi[k] = \phi(Y_{w,(k-\tilde{N},k]}) - y_w[k]$ and $\varphi[k] \neq 0$ only for $k_a \leq k < k_e$, with k_e the attack end time.

The effect on y_w will translate, through the remover $\mathcal{Q}$, into an effect on the reconstructed output y_q which, if not detected promptly, could cause performance degradation or catastrophic failures as y_q is used by the controller $\mathcal{C}$ to compute its control action for the plant $\mathcal{P}$. To analyze the effect on y_q and the conditions under which this can be detected by $\mathcal{R}$, we will separately analyze the following cases during $k_a \leq k < k_e$, which arise as a consequence of the switching protocol we designed into $\mathcal{W}$ and $\mathcal{Q}$.

1) $(\tau_w, \tau_q) = (0, 0)$: no switch is triggered at $\mathcal{W}$ and at $\mathcal{Q}$,
2) $(\tau_w, \tau_q) = (0, 1)$: a switch is triggered at $\mathcal{Q}$ but not at $\mathcal{W}$,
3) $(\tau_w, \tau_q) = (1, 0)$: a switch is triggered at $\mathcal{W}$ but not at $\mathcal{Q}$,
4) $(\tau_w, \tau_q) = (1, 1)$: a switch is triggered at both $\mathcal{W}$ and $\mathcal{Q}$.

In the analysis, we assume that $\mathcal{W}$ and $\mathcal{Q}$ were synchronized at instant $k_a - 1$. Cases 1) and 4) correspond to situations where $\mathcal{W}$ and $\mathcal{Q}$ are still switch synchronized, albeit they are generally not synchronized in the sense of Definition 4 as their states would be different because of the attack. Cases 2) and 3), instead, depict instances where $\mathcal{W}$ and $\mathcal{Q}$ are not even trigger-synchronized during and possibly after the attack.

1) Case $(\tau_w, \tau_q) = (0, 0)$: In this case, as there are no switches at $\mathcal{W}$ and at $\mathcal{Q}$ during the attack period, we can write the following expression for the attacked state $\tilde{x}_{q1}$ of $\mathcal{Q}$:

$$\begin{aligned} \tilde{x}_{q1}[k] &= \sum_{h=k_a}^{k-1} \left(A_q\right)^{k-1-h} B_q\left(y_w[h] + \varphi[h]\right) \\ &\quad + \left(A_q\right)^{k-k_a} \times x_q[k_a] \\ &= x_q[k] + \sum_{h=k_a}^{k-1} \left(A_q\right)^{k-1-h} B_q\varphi[h]. \end{aligned} \tag{26}$$

From (26), it follows that the attacked reconstructed output can be expressed as

$$\tilde{y}_{q1}[k] = y_p[k] + C_q \sum_{h=k_a}^{k-1} (A_q)^{k-1-h} B_q \varphi[h] + D_q \varphi[k] \\ \triangleq y_p[k] + \varphi_{q1}[k] \tag{27}$$

where φ_{q1} is the result of filtering the attack φ through the remover $\mathcal{Q}$ in the present case. It is now possible to derive the following result on the attack detectability.

Theorem 5 (Attack Detectability Under no Switch Conditions): If there exists a time index $k_d > k_a$ and a component $i \in \{1, \dots, n_y\}$ such that during a MITM attack the following inequality holds:

$$\left| C_{p,(i)} \left[\sum_{h=k_a}^{k_d-1} (A_r)^{k_d-1-h} (-K\varphi_{q1}[h]) \right] + \varphi_{q1}[k_d] \right| \\ > 2\nu_1^i \sum_{h=0}^{k_d-1} \left(\nu_2^i\right)^{k_d-1-h} \left(\bar{\eta}[h] + \|K\|\bar{\xi}[h]\right) \\ + \left(\nu_2^i\right)^{k_d-k_a} \left(\nu_1^i \bar{\epsilon}[k_a] + \bar{y}_{r,(i)}[k_a]\right) + 2\bar{\xi}[k_d] \tag{28}$$

then the attack will be detected at the time instant k_d.

Proof: During a MITM attack, the solution for the output residual y_r during an attack can be computed using the same approach, we used for (23), leading to

$$y_r[k] = C_p \left[\sum_{h=k_a}^{k-1} (A_r)^{k-1-h} (-K(\xi[h] + \varphi_{q1}[h]) \\ + \eta[h]) + (A_r)^k \epsilon[k_a] \right] + \varphi_{q1}[k] + \xi[k] \, .$$

The proof then follows from Definition 1 and [23, Th. 3.1]. ■

The assumption on the absence of switches translates, considering Lemma 3, to the assumed condition $\bar{\delta}_{y_q} \triangleq \max_{k_a \le k < k_e} |y_q[k] - \beta| \le \delta$. This means, thus, that a sufficient condition on the attack amplitude for meeting the assumption that there are no switches shall be $|\varphi_{q1}[k]| \le \delta - \bar{\delta}_{y_q} \triangleq \bar{\varphi}_{q,1}$.

For the particular case of the switching protocol of Section III-C, the term β is equal to $y_q[k-1]$ and a necessary condition on φ_{q1} is easily written as $|y_q[k] + \varphi_{q1}[k] - y_q[k-1] - \varphi_{q1}[k-1]| \le \delta$. The worst case, from the point of view of the magnitude of the attack, occurs when $y_q[k] - y_q[k-1] = \pm\delta$ to which corresponds $\varphi_{q1}[k] - \varphi_{q1}[k-1] = \mp 2\delta$. By looking at Theorem 5, we can see that, for a fixed observer gain K, the amplitude of the attack signal φ_{q1} should be large enough to overcome the effect of the uncertainty terms in the right-hand side of the hypothesis in order to have detection. As the deleterious effects of an attack are dependent on its magnitude, it means that potentially more dangerous attacks are more likely to be detected, while smaller ones will not. Anyway, if the constant δ in the switching protocol is chosen small enough, we can make so that even small attacks will trigger a switch at $\mathcal{Q}$, which will ease detection due to the loss of synchronicity and mismatch between $\mathcal{W}$ and $\mathcal{Q}$, as analyzed in the next two cases.

2) Case $(\tau_w, \tau_q) = (0, 1)$***:*** Suppose that at time $\tilde{\kappa}_{q2}$, with $k_a \le \tilde{\kappa}_{q2} < k_e$, the attack value $\varphi[\tilde{\kappa}_{q2}]$ is high enough to cause a switch at $\mathcal{Q}$, but no switches occur at $\mathcal{W}$ during the period $k_a \le k < k_e$. We also assume no attack was detected during the preswitch period $k_a \le k < \tilde{\kappa}_{q2}$, otherwise case 1 would have applied. Indeed, during the preswitch period, the attacked state $\tilde{x}_{q2}$ and output $\tilde{y}_{q2}$ follow the same expressions as those for case 1. At switch time $\tilde{\kappa}_{q2}$, instead, it holds

$$\begin{cases} \tilde{\theta}_q[\tilde{\kappa}_{q2}] & = \sigma_q(\theta_q[k_a]) \\ \tilde{x}_{q2}^{+}[\tilde{\kappa}_{q2}] & = \rho_q(\tilde{x}_{q2}^{-}[\tilde{\kappa}_{q2}], \tilde{y}_w[\tilde{\kappa}_{q2}], \theta_q[k_a], \tilde{\theta}_q^{+}[\tilde{\kappa}_{q2}]) \\ \varphi_{q2}[\tilde{\kappa}_{q2}] & = \alpha^{-1}(\hat{\alpha}(\tilde{y}_w[\tilde{\kappa}_{q2}])) - y_p[\tilde{\kappa}_{q2}] \end{cases} \tag{29}$$

where the term φ_{q2} denotes the effect of the attack on the reconstructed output and $\tilde{\theta}_{q2}$ denotes $\mathcal{Q}$ parameter during the attack in this case. It is thus interesting to notice that that initial effect of the attack on $\tilde{y}_{q2}$ depends also on the sensitivity of the composite function $\alpha^{-1}(\hat{\alpha})$ on its argument.

During the postswitch period $\tilde{\kappa}_{q2} \le k < k_e$, the reconstructed output solution is equal to

$$\tilde{y}_{q2}[k] = C_q(\tilde{\theta}_{q2}) \left[\sum_{h=\tilde{\kappa}_{q2}}^{k-1} \left(A_q(\tilde{\theta}_q)\right)^{k-1-h} B_q(\tilde{\theta}_q) \left(\tilde{y}_w[h]\right)\right) \\ + \left(A_q(\tilde{\theta}_q)\right)^{k-\tilde{\kappa}_{q2}} \tilde{x}_{q2}[\tilde{\kappa}_{q2}] \right] + D_q(\tilde{\theta}_q) \left(\tilde{y}_w[k]\right) \, . \tag{30}$$

We can now state the following result.

Corollary 2 (Attack Detectability Under Extra $\mathcal{Q}$ Switch Conditions): Let us assume an attack φ is affecting y_w during the period $k_a \le k < k_e$ and it is such that $\tau_q(\tilde{y}_w[k]) = 1$ for $k = \tilde{\kappa}_{q2}$ and 0 otherwise. If $\tau_w(y_p[k]) = 0$ for all $k \in [k_a, \, k_e]$ and condition (28) holds at a time instant $k_d \in [\tilde{\kappa}_{q2}, \, k_e]$ with the term

$$\varphi_{q2}[k] = \mathcal{D}(\theta_w, \tilde{\theta}_q) y_p[k] + \mathcal{Q}(\tilde{\theta}_q)\varphi[k] \\ + C_q(\tilde{\theta}_q) A_q(\tilde{\theta}_q)^{k-\tilde{\kappa}_{q2}} \tilde{x}_{q2}[\tilde{\kappa}_{q2}] \\ - C_q(\theta_q) A_q(\theta_q)^{k-\tilde{\kappa}_{q2}} x_q[\tilde{\kappa}_{q2}] \tag{31}$$

placed in lieu of term φ_{q1}, then the attack will be detected at k_d.

Proof: It follows directly from Theorem 5, if φ_{q2} is computed by taking the difference between the solution for $y_q[k]$ in nonattacked conditions and the expression in (30). By remembering the definition of $\mathcal{D}$ as the system introduced in (9) and setting null initial conditions for both $\mathcal{D}(\theta_w, \tilde{\theta}_q)$ and $\mathcal{Q}(\tilde{\theta}_q)$, the thesis is obtained. ■

3) Case $(\tau_w, \tau_q) = (1, 0)$***:*** This case is indeed similar to the previous one, with the difference that now we assume the effect of the attack is to hide from $\mathcal{Q}$ a switch occurring at $\mathcal{W}$. This means that $\tau_w(y_p[\kappa_w]) = 1$ and $\tau_q(y_w^{+}[\kappa_w] + \varphi[\kappa_w]) = 0$ at the switch time $\kappa_w \in [k_a, \, k_e]$. For the period $[k_a, \, \kappa_w]$, up to and including the switch time κ_w, the attacked state $\tilde{x}_{q3}$ solution is the same as $\tilde{x}_{q1}$ for case 1, and again no detection is assumed to occur. At the switch time, the effect on y_q of the attack is described by the term φ_{q3}

$$\begin{aligned}\varphi_{q3}[\kappa_w] = C_q(\tilde{\theta}_q[\kappa_w])\tilde{x}_q[\kappa_w] + D_q(\tilde{\theta}_q[\kappa_w])(y_w^+[\kappa_w]\\ + \varphi[\kappa_w])) - \alpha^{-1}(\hat{\alpha}(y_w^+[\kappa_w]))\end{aligned} \tag{32}$$

where for the attacked parameter vector it will hold $\tilde{\theta}_q[\kappa_w] = \theta_q[k_a]$ as there had been no switches at $\mathcal{Q}$. The subsequent evolution of $\mathcal{Q}$ will follow (5) without any switches. It is then straightforward to see that, provided these different initial conditions at κ_w, the same result as in Corollary 2 holds, and will thus not be repeated.

4) Case** $(\tau_w, \tau_q) = (1, 1)$**: In this case, we assume a switch is triggered both at $\mathcal{W}$ and at $\mathcal{Q}$ at time $\kappa_w = \kappa_q \in [k_a,\ k_e]$, i.e., that the attack is not able to hide from $\mathcal{Q}$ the switch. The solution of the attacked state $\tilde{x}_{q4}$ during the period $[k_a,\ \kappa_w]$ will be the same as in case 1 and the attack presence will lead $\mathcal{Q}$ to compute a wrong reset output and state. This is captured by the following equation valid at time κ_w

$$\begin{cases}\theta_q^+[\kappa_w] &= \sigma_q(\theta_q[k_a]) = \theta_w^+[\kappa_w]\\ \tilde{x}_{q4}^+[\kappa_w] &= \rho_q(\tilde{x}_{q4}^-[\kappa_w], \tilde{y}_w[\kappa_w], \theta_q[k_a], \theta_q^+[\kappa_w])\\ \varphi_{q4}[\kappa_w] &= \alpha^{-1}(\hat{\alpha}(y_w^+[\kappa_w] + \varphi[\kappa_w])) - \alpha^{-1}(\hat{\alpha}(y_w^+[\kappa_w]))\,.\end{cases}$$

The generator $\mathcal{W}$ and the remover $\mathcal{Q}$ are thus switch synchronized, but not synchronized at time instant κ_w.

By applying similar reasoning as in cases 1 and 2, we can enunciate the following.

Corollary 3 (Attack Detectability Under Switch Synchronization Conditions): Let us assume an attack φ is affecting y_w during the period $k_a \le k < k_e$ and it is such that $\tau_w(y_p[\kappa_w]) = \tau_q(y_w[\kappa_w] + \varphi[\kappa_w]) = 1$ for $\kappa_w \in [k_a,\ k_e]$. If condition (28) holds at a time instant $k_d \in [\kappa_w,\ k_e]$ with

$$\varphi_{q4}[k] = \mathcal{Q}(\theta_q)\varphi[k] + C_q(\theta_q)A_q(\theta_q)^{k-\kappa_w}\,(\tilde{x}_{q4}[\kappa_w] - x_q[\kappa_w]) \tag{33}$$

placed in lieu of φ_{q1}, then the attack will be detected at k_d.

Proof: It follows directly from Corollary 2, by noting that $\theta_q = \theta_w$ holds under switch synchronization and thus $\mathcal{D} = 0$. ■

Remark 5: As can be seen in the previous analyses, only when no switch occurs during an attack (case 1) does the detectability depend uniquely on the attack magnitude. In the other three cases, when at least one switch occurs, then the detectability is also influenced by the mismatch between $\mathcal{W}$ and $\mathcal{Q}$ parameters and/or states. This suggests that, in order to improve attack detectability, the switching protocol must be designed in a way that during an attack there will be a switch with high probability.

To give an insight into this and better relate the effect of one specific kind of attack to the general theory developed so far, we are presenting next an illustrative example and some numerical results.

VII. Numerical Examples

In this section, we will present a numerical analysis of the effects of two cases of stealthy MITM attack and of how the proposed watermarking scheme can detect it, as a mean to show its effectiveness. In particular, we will consider here the two cases of replay and data injection attacks occurring to a second order, unstable LTI system $\mathcal{P}$, which was first introduced as a test case in [17].

A. Closed-Loop System

The plant, which may represent for instance a second-order mechanical system linearized around an unstable equilibrium point, comprises two states, one input and two outputs, and its state-space matrices are the following:

$$A_p = \begin{bmatrix} 1 & 0.1 \\ 0.035 & 0.99 \end{bmatrix},\ B_p = \begin{bmatrix} 0 \\ 1 \end{bmatrix},\ C_p = I_2$$

where I_2 denotes the 2×2 identity matrix. The sampling step has been set equal to $T_s = 0.1$ s, while the eigenvalues of A_p results equal to 1.0144 and 0.9756. The controller has been designed via pole placement and is represented by the following state-space matrices $A_c = I_2$, $B_c = 0.1 \cdot I_2$, $C_c = [0.01\ 0.022]$, $D_c = [0.0875\ 0.1980]$.

In order to be able to implement a stealthy replay attack, we will assume that the system $\mathcal{P}$ is operated as a periodic batch process, and that the attack will record exactly one period of the batch and replay it perfectly in sync with the next one. To this end, we will introduce a periodic reference signal r: in particular, $r_{(1)}$ will be chosen to be a square wave varying between 0.5 and 1.5 and having a period of 100 s, while $r_{(2)}$ will be a null signal. The control error $e \triangleq r - y_q$ between the reference and the reconstructed output from $\mathcal{Q}$ will thus be fed into the controller. We will assume that the model uncertainty η will be caused by a parametric uncertainty in the model of $\mathcal{P}$ used to implement the observer of the detector $\mathcal{R}$, such uncertainty being random and no higher than 5 % of the nominal values. On top of this, the measurement uncertainty ξ will be implemented as a random variable uniformly distributed in the interval $[-0.025\ 0.025]^2$.

B. Replay Attack

The attacker is assumed to start recording both sensor outputs at time $T_{\text{REC}} = 40$ s and start to replay them 100 s later, at time $T_a = k_a \cdot T_s = 140$ s. As anticipated this choice is making the attack stealthy, as the replayed data is identical, except for the measurement noise, to the data the plant would have produced in the attack absence. We can indeed appreciate this by plotting the simulation results where the proposed watermarking is implemented with identities, that is both $\mathcal{W}$ and $\mathcal{Q}$ pass through their inputs unaltered as if the scheme were not in place.

As shown in Fig. 3, due to $\mathcal{P}$ being unstable and the replay attack being such to open the feedback loop when active, immediately after the attack start time T_a the true plant outputs diverge, these being plot as a solid blue line in the figure. Anyway, the values received at the network side where $\mathcal{Q}$, $\mathcal{C}$, and $\mathcal{R}$ are situated seem from any point of view the kind of data we would have expected. These data, plotted as yellow circular markers, seem perfectly in line with the data from the first period that started at 0 s and lasted until 100 s. A look at the residuals produced by the detector does not show anything suspicious either (Fig. 4), and indeed both residuals remain below their respective thresholds.

The introduction of nontrivial watermarking filters is making the attack detectable, as will be illustrated next. For the

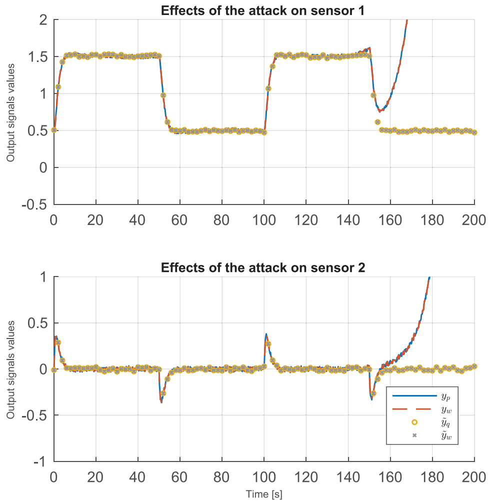

Fig. 3. Plant outputs during a replay attack (starting at 140 s), when the watermarking scheme is not implemented.

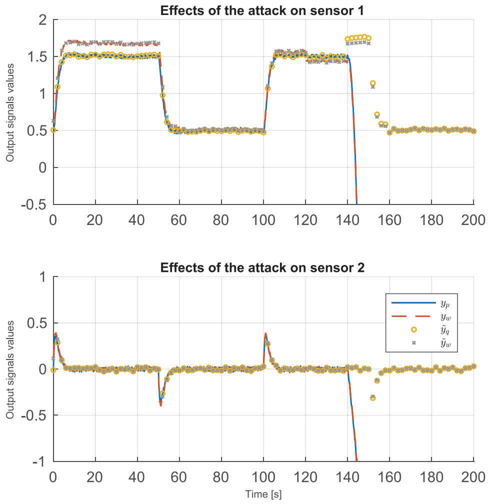

Fig. 5. Plant outputs during a replay attack (starting at 140 s), when the watermarking scheme is implemented according to Section III-C.

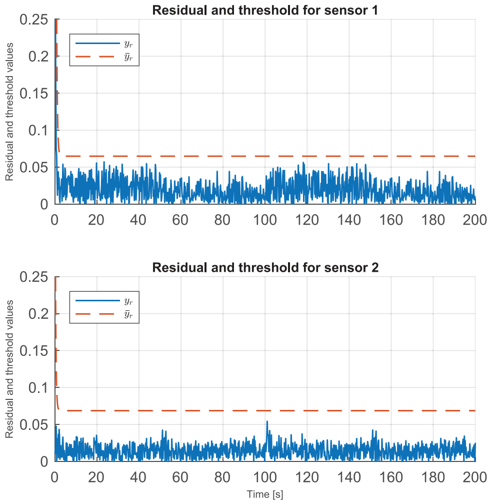

Fig. 4. Residuals and thresholds during a replay attack (starting at 140 s), when the watermarking scheme is not implemented.

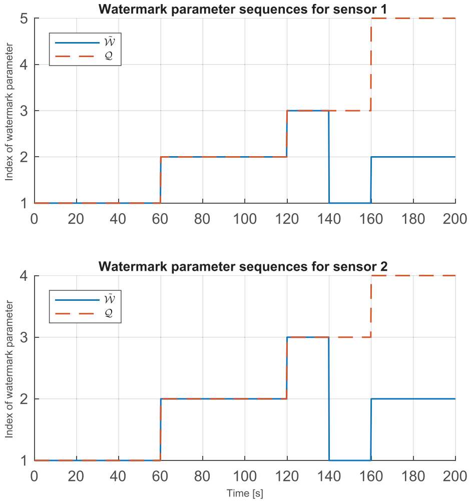

Fig. 6. Index of watermarking parameters during a replay attack (starting at 140 s), when the watermarking scheme is implemented according to Section III-C. In the legend, a tilde over W refers to the index that was used for the generator replayed data.

watermark generator, a sequence of five different fourth order finite impulse response (FIR) filters will be generated randomly. In particular, the coefficients will be chosen according to the equation $w_B^\top = [1,\ 0,\ 0,\ 0] + \omega$, with ω being a random vector whose components are each uniformly distributed in the box $[-0.075\ 0.075]$.

The resulting coefficients not only guarantee the stability of the corresponding equalizer filter, but will also respect the stability condition of Theorem 4 for the present system $\mathcal{P}$. Every vector of filter coefficients will constitute one possible value of the parameters θ_w and θ_q, and the switch functions σ_w and σ_q will simply circularly select the next one in the sequence. To this end, the triggering function τ_w will be defined in a way to trigger a direct switch at time instants corresponding to 60, 120, and 180 s.

As a synchronization protocol, we will implement the example presented in Section III-C, with a value $\delta = 0.25$. By examining Fig. 5, it is possible to see how, before the attack starts at 140 s, the signal y_w is indeed different than y_p because of the watermarking, but the output y_q is reconstructing y_p exactly. Furthermore, it is possible to notice that during the attack y_q is no longer looking like the previous periods of the batch, but is experiencing noticeable jumps, in correspondence of the switches of $\mathcal{Q}$.

The switches are plotted in Fig. 6, alongside those of the replayed signal y_w, by indicating the current sequence index of the parameter θ_w and θ_q used. From this figure, it is evident how synchronization was kept before the start of the attack, but it is lost right at 140 s.

The effect of the loss of synchronization, due to the replay attack, is even more evident from the plot of the residuals for the two outputs (see Fig. 7). In order to appreciate these plots, we need first to determine, from the point of the analyses carried out in Section VI, how a replay attack translates into the attack signal φ defined in (25). In this case, the attribution is simple, as the attack is perfectly synchronized with the periodicity of the

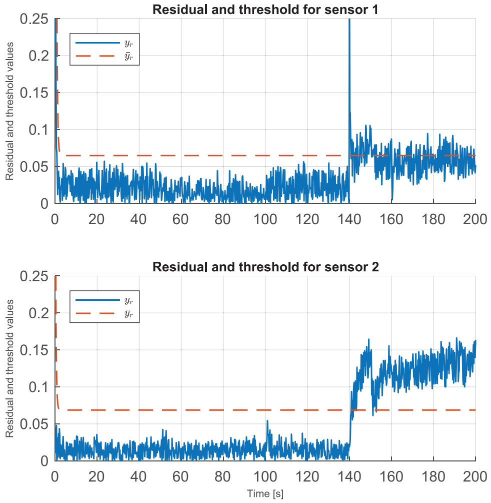

Fig. 7. Residuals and thresholds during a replay attack (starting at 140 s), when the watermarking scheme is implemented according to Section III-C.

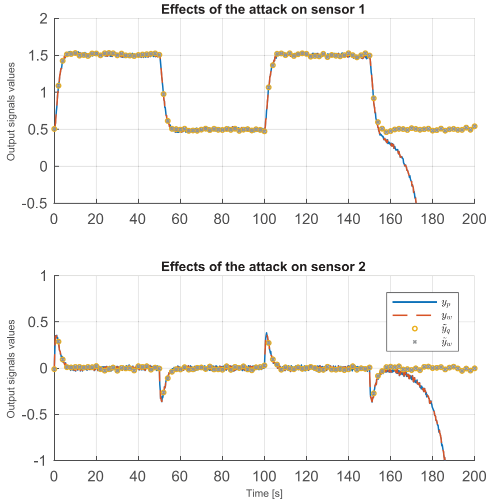

Fig. 8. Plant outputs during a data injection attack (starting at 140 s), when the watermarking scheme is not implemented.

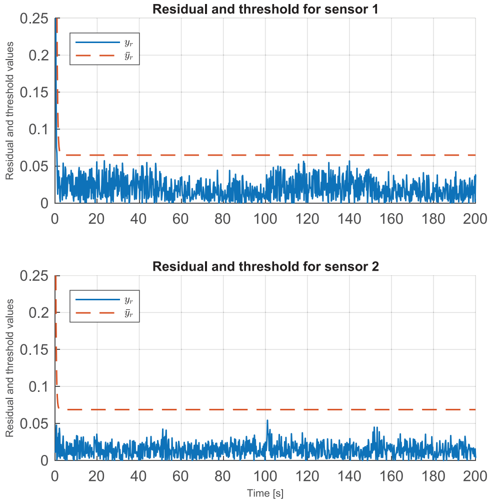

Fig. 9. Residuals and thresholds during a data injection attack (starting at 140 s), when the watermarking scheme is not implemented.

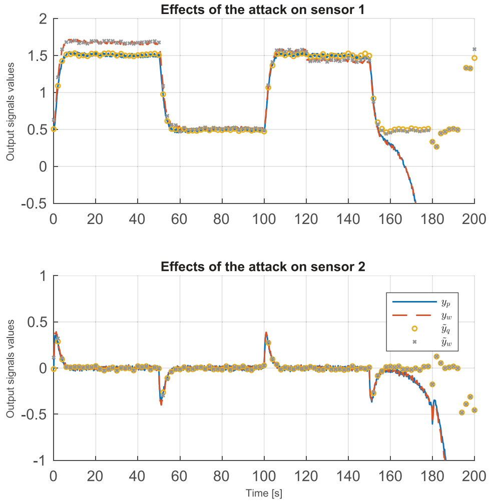

Fig. 10. Plant outputs during a data injection attack (starting at 140 s), when the watermarking scheme is implemented as in the previous example.

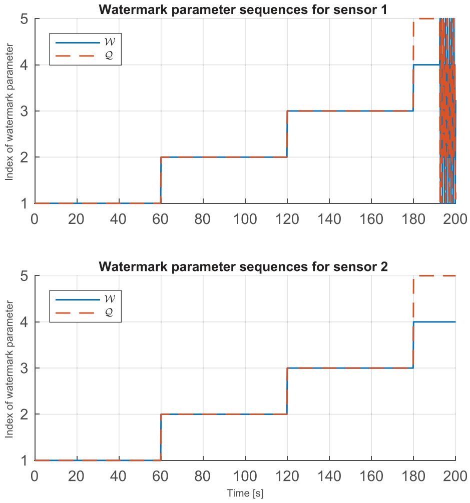

Fig. 11. Index of watermarking parameters during a data injection attack (starting at 140 s), when the watermarking scheme is implemented as in the previous example .

process. So, φ turns out to be the difference between the effect of the process and measurement noises at recording time and at replaying time.

Furthermore, for this interpretation to be valid, we need to assume that at the start of the replay the parameter θ_w instantly changes from the actual value to the value it had at the start of the recording time. Given this interpretation, we can easily realize that the start of the replay attack corresponds to case 3 in the analysis of Section VI-A. Indeed at time T_a, we can notice, especially for the first sensor, a spike in the residual which is due to the mismatch in the reset conditions of $\mathcal{W}$ and $\mathcal{Q}$, as could have been expected from the condition in Theorem 5 and Corollary 2. The value of the residual is also kept high by the second term in the right-hand side of (32), which are less evident in the period from 150 to 200 s as the plant output has a lower value there.

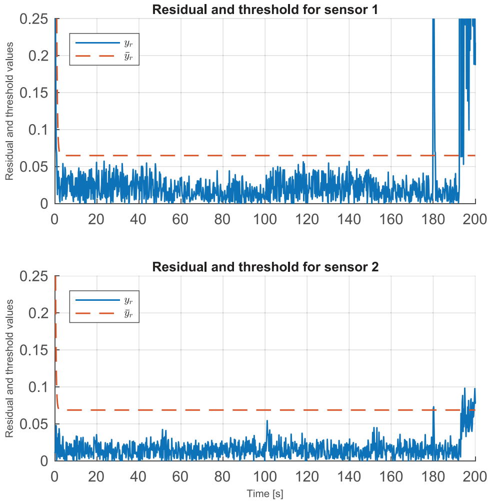

Fig. 12. Residuals and thresholds during a data injection attack (starting at 140 s), when the watermarking scheme is implemented as in the previous example.

Both the threshold for sensor 1 and for sensor 2 are crossed right after the attack start time, thus verifying the capability of the proposed scheme to identify MITM attacks that would have been otherwise stealthy.

C. Data Injection Attack

As a second example, we did consider the same stealthy data injection attack introduced in [16]. In particular, at time $T_a = k_a \cdot T_s = 140$ s, a measurement false-data injection attack described by $\varphi[k] = C_p A_p^{k-k_a} x_a = \lambda^{k-k_a} C_p x_a$, with $x_a = 10^{-2}[0.9898\ 0.1422]^\top$ and $\lambda = 1.0144$, starts to excite the plant's unstable mode.

When no watermarking is used (Figs. 8 and 9), the exponentially increasing attack signal causes the true plant output y_p to quickly diverge, while the estimated output $\hat{y}_p$ appears to follow the square wave reference faithfully. Similarly, the residual and threshold, do not reveal any sign of the attack.

The introduction of the watermarking scheme proposed here and implemented as in the case of the replay attack example, leads instead to a quick detection as soon as a switch occurs (Figs. 10–12). Finally, we would like to point out that, in both examples, no false alarms are raised during the attack-free period from 0 to 140 s, as is expected from Lemma 7.

VIII. Conclusion

Inspired in authentication techniques with weak cryptographic guarantees, we have proposed a multiplicative watermarking scheme for networked control systems. In this scheme, the sensors' outputs are fed to a watermark generator, which produces the watermarked data that is transmitted through the possibly unsecured communication network. At the controller's side, the watermark remover reconstructs the original measurement data.

Design guidelines for choosing the filter parameters are provided. Specifically, we derive design rules, which ensure that, in the absence of attacks, the nominal closed-loop system performance is not deteriorated by the watermarking scheme. Moreover, we show that the filters must change their parameters sufficiently often to detect replay attacks. Modeling the filters as hybrid discrete-time systems, we design a switching protocol with no communication overhead to allow the watermark generator and remover to synchronously update their filter parameters. Furthermore, the stability of the system with mismatched parameters is analyzed. As a result, we identify tradeoffs that must be considered between the robust stability of the closed-loop system and the detectability of attacks. The results are illustrated through numerical examples, which verify the effectiveness of the approach.

As future work, extensions to the actuator case are envisioned, as well as the applicability of the approach to other attack scenarios. Additionally, we plan to further analyze the global stability of the system without synchronization, and to investigate the complete filter design problem to maximize detectability while ensuring robust stability. Finally, extensions to nonlinear watermark generators or systems are of high practical and theoretical interest, as well as the application of the proposed approach to realistic physical benchmarks as that described in [24].

Appendix

Proof of Lemma 6: First observe that $|y_w^+[\kappa_w] - \beta| = \max\{|\alpha - \beta|,\ \delta + \delta_p\}$. Given the properties of δ_p, we can derive $|\alpha - \beta| = \min\{|y_w^+[\kappa_w] - \beta|,\ \delta_p^{-1}(|y_w^+[\kappa_w] - \beta| - \delta)\}$, where we assign $\delta_p^{-1}(x) = +\infty$ for $x < 0$. In other words, we may compute the quantity $|\alpha - \beta|$ given β, $y_w^+[\kappa_w]$, δ, and the function $\delta_p^{-1}(x)$, which are all available at the remover. The proof follows by considering two cases.

First suppose that $|y_w^+[\kappa_w] - \beta| \leq \delta_p^{-1}(|y_w^+[\kappa_w] - \beta| - \delta)$, which corresponds to the first case in (18). In this case, we have $|y_w^+[\kappa_w] - \beta| = |\alpha - \beta|$ and, given the characterization of $y_w^+[\kappa_w]$ in (17), we further conclude that $\alpha = y_w^+[\kappa_w]$.

The second case occurs when $|y_w^+[\kappa_w] - \beta| > \delta_p^{-1}(|y_w^+[\kappa_w] - \beta| - \delta)$ holds, for which we have $|\alpha - \beta| = \delta_p^{-1}(|y_w^+[\kappa_w] - \beta| - \delta)$. To compute α in this case, one needs to further know the sign of $\alpha - \beta$. Given that g is an even function with respect to α, we observe that $\text{sign}(\alpha - \beta) = \text{sign}(y_w^+[\kappa_w] - \beta)$, and thus α can be obtained as $\alpha = \beta + \text{sign}(y_w^+[\kappa_w] - \beta)\delta_p^{-1}(|y_w^+[\kappa_w] - \beta| - \delta)$, which concludes the proof. ■

Proof of Theorem 3: First, we address the asymptotic stability of the closed-loop system. Under synchronization at all times, invoking Theorem 1 directly concludes the proof: the trajectories of $x_p[k]$, $x_{cr}[k]$, and $y_p[k]$ with the watermarking scheme coincide with the nominal trajectories without watermarking, thus these variables converge asymptotically to the origin.

The second part of the proof focuses on the internal states of the generator and remover, $x_w[k]$ and x_q[k], respectively. Due to the parameter switching scheme, these states are reset at switching times, which prevents them from converging to the origin. Instead, as we show next, these states converge asymptotically to a region around the origin, that is, they are uniformly ultimately bounded.

First, we investigate the behavior at switching times κ_i. Recall the state jump at switching times, (19), and denote $x_w[\kappa_i]$ as its optimal solution. Define $x'_w = \frac{C_w^\top}{\|C_w\|^2}(y_w^+[\kappa_i] - D_w y_p[\kappa_i])$, which is a feasible solution to (19). Thus, recalling that $h(x) = \|x\|$ and that the trajectory of y_p is decoupled from that of x_w, we have the relations

$$\begin{aligned}\|x_w[\kappa_i]\| \le \|x'_w\| &= \frac{1}{\|C_w\|}|y_w^+[\kappa_i] - D_w y_p[\kappa_i]| \\ &\le \frac{1}{\|C_w\|}|f(0)| + M\sigma^{\kappa_i}\end{aligned}$$

where $\sigma \in (0,1)$ is the decay rate of y_p and $M = \frac{1}{\|C_w\|}(-|f(0)| + \sup_k |f(y_p[k]) - D_w y_p[k]|)$.

The remaining case is the behavior at interswitching time-intervals. Since the generator and remover are both stable during interswitching intervals, their states decay asymptotically toward zero during time intervals in between switching instants. In other words, for all $k \in [\kappa_i,\ \kappa_{i+1})$, there exist positive constants $M_i < \infty$ and $\sigma_i \in (0,1)$ such that $\|x_w[k]\| \le M_i \sigma_i^{k-\kappa_i}$, which concludes the proof. ■

References

[1] Trend Micro, "Unseen threats, imminent losses 2018 midyear security roundup," 2018. [Online]. Available: https://documents.trendmicro.com/assets/rpt/rpt-2018-Midyear-Security-Roundup-unseen-threats-imminent-losses.pdf

[2] B. Gorenc and F. Sands, "The state of SCADA HMI vulnerabilities," 2018. [Online]. Available: https://documents.trendmicro.com/assets/wp/wp-hacker-machine-interface.pdf

[3] NCCIC and ICS-CERT, "ICS-CERT year in review 2016," 2016. [Online]. Available: https://ics-cert.us-cert.gov/sites/default/files/Annual_Reports/Year_in_Review_FY2016_Final_S508C.pdf

[4] A. A. Cárdenas, S. Amin, and S. S. Sastry, "Secure control: Towards survivable cyber-physical systems," in *Proc. 28th Int. Conf. Distributed Comput. Syst. Workshops*, 2008, pp. 495–500.

[5] A. A. Cárdenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, and S. S. Sastry, "Challenges for securing cyber physical systems," in *Proc. Workshop Cyber-Phys. Syst. Secur.*, Jul. 2009.

[6] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," *Automatica*, vol. 51, no. 1, pp. 135–148, 2015.

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

[8] R. Smith, "A decoupled feedback structure for covertly appropriating networked control systems," in *Proc. 18th IFAC World Congr.*, 2011, pp. 90–95.

[9] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "Revealing stealthy attacks in control systems," in *Proc. 50th Annu. Allerton Conf. Commun., Control, Comput.*, 2012, pp. 1806–1813.

[10] F. Miao, Q. Zhu, M. Pajic, and G. J. Pappas, "Coding schemes for securing cyber-physical systems against stealthy data injection attacks," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 1, pp. 106–117, Mar. 2017.

[11] L. Pérez-Freire, P. Comesaña, J. R. Troncoso-Pastoriza, and F. Pérez-González, *ransactions on Data Hiding and Multimedia Security I*, in , ch. Watermarking Security: A Survey. Berlin, Germany: Springer 2006.

[12] B. Satchidanandan and P. R. Kumar, "Dynamic watermarking: Active defense of networked cyber–physical systems," *Proc. IEEE*, vol. 105, no. 2, pp. 219–240, 2016.

[13] Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs," *IEEE Control Syst. Mag., IEEE*, vol. 35, no. 1, pp. 93–109, Feb. 2015.

[14] A. J. Gallo, M. S. Turan, F. Boem, G. Ferrari-Trecate, and T. Parisini, "Distributed watermarking for secure control of microgrids under replay attacks," *IFAC-PapersOnLine*, vol. 51, no. 23, pp. 182–187, 2018.

[15] S. Weerakkody and B. Sinopoli, "Detecting integrity attacks on control systems using a moving target approach," in *Proc. 54th IEEE Conf. Decis. Control*, Osaka, Japan, Dec. 2015, pp. 3757–3764.

[16] R. M. Ferrari and A. H. Teixeira, "Detection of sensor data injection attacks with multiplicative watermarking," in *Procs. Eur. Control Conf.*, 2018, ppp. 338–343.

[17] R. M. Ferrari and A. H. Teixeira, "Detection and isolation of replay attacks through sensor watermarking," in *Procs. 20th IFAC World Congr.*, 2017.

[18] R. M. G. Ferrari and A. M. H. Teixeira, "Detection and isolation of routing attacks through sensor watermarking," in *Proc. Amer. Control Conf.*, May 2017, pp. 5436–5442.

[19] W. Knowles, D. Prince, D. Hutchison, J. F. P. Disso, and K. Jones, "A survey of cyber security management in industrial control systems," *Int. J. Critic. Infrastruct. Prot.*, vol. 9, pp. 52–80, 2015.

[20] R. Goebel, R. G. Sanfelice, and A. R. Teel, "Hybrid dynamical systems," *IEEE Control Syst. Mag.*, vol. 29, no. 2, pp. 28–93, Apr. 2009.

[21] A. R. Teel and J. I. Poveda, "A hybrid systems approach to global synchronization and coordination of multi-agent sampled-data systems," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 123–128, 2015.

[22] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1996.

[23] R. M. Ferrari, T. Parisini, and M. Polycarpou, "A robust fault detection and isolation scheme for a class of uncertain input-output discrete-time nonlinear systems," in *Proc. Amer. Control Conf.*, Jun. 2008, pp. 2804–2809.

[24] A. P. Mathur and N. O. Tippenhauer, "SWaT: A water treatment testbed for research and training on ICS security," in *Proc. Int. Workshop Cyber-Phys. Syst. Smart Water Netw.*, 2016, pp. 31–36.

Riccardo M. G. Ferrari received the Laurea degree (Cum Laude and printing hons.) in electronic engineering and the Ph.D. degree in information engineering, both from University of Trieste, Trieste, Italy, in 2004 and 2009, respectively.

He held both academical and industrial R&D positions, in particular as Researcher in the field of process instrumentation and control for the steel-making sector. He is a Marie Curie alumnus and currently an Assistant Professor with the Delft Center for Systems and Control, Delft University of Technology, The Netherlands. His research interests include wind power fault tolerant control and fault diagnosis and attack detection in large-scale cyber-physical systems, with applications to electric vehicles, cooperative autonomous vehicles and industrial control systems.

Dr. Ferrari was the recipient of the 2005 Giacomini Award of the Italian Acoustic Society and he obtained the second place in the Competition on Fault Detection and Fault Tolerant Control for Wind Turbines during IFAC 2011. Furthermore, he was also the recipient of an Honorable Mention for the Pauk M. Frank Award at the IFAC SAFEPROCESS in 2018 and won an Airbus Award at IFAC 2020 for the best contribution to the competition on Aerospace Industrial Fault Detection.

André M. H. Teixeira received the M.Sc. degree in electrical and computer engineering from the Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, in 2009, and the Ph. D. degree in automatic control from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2014.

He is currently an Associate Senior Lecturer with the Division of Signals and Systems, Department of Electrical Engineering, Uppsala University, Uppsala, Sweden. From 2015 to 2017, André was an Assistant Professor in cybersecurity of critical infrastructures with the Faculty of Technology, Policy and Management, Delft University of Technology, The Netherlands. His current research interests include secure and resilient control systems, distributed fault detection and isolation, distributed optimization and power systems.

Dr. Teixeira was a recipient for the Best Student-Paper Award from the IEEE Multiconference on Systems and Control in 2014 and an Honorable Mention for the Pauk M. Frank Award at the IFAC SAFEPROCESS in 2018. He was awarded a Starting Grant by the Swedish Research Council in 2019, and he is among the 20 young researchers in Sweden that received the Future Research Leaders 7 grant by the Swedish Foundation for Strategic Research in 2020.