Stabilization of linear cyber-physical systems against attacks via switching defense

Shenyu Liu   Sonia Martínez   Jorge Cortés

Abstract—**This paper studies cyber-physical systems modeled with linear dynamics subject to attacks on its parameters. The attacker knows at all times the defense employed and injects a destabilizing piecewise Lipschitz time-varying attack signal. The defender does not know the specific attack and aims to preserve system stability. We propose a partitioning strategy for the set of possible attacks that generates a finite collection of candidate defenses such that, for each member of the partition, there is a defense that stabilizes the system with respect to all static attack signals belonging to it. The defender then implements a mechanism that switches among the candidate defenses based on the evaluation of a Lyapunov-based criterion that determines whether the current defense is stabilizing. We characterize the properties of the switched time-varying system with delay, the latter arising from the interval between the switch-triggering events and their actual implementation. Our analysis provides a tolerance on the implementation delay that prevents the defense signal from constantly switching. In addition, we also identify a condition on the switching frequency that ensures global exponential stability. Simulations of the proposed switched defense mechanism illustrate its performance and advantages over static defenses.**

I. Introduction

Cyber-physical systems (CPSs) require the tight integration of multiple facets, including computation, communication, and control, to operate reliably and efficiently in the physical world. CPSs are widespread in many important application domains such as industrial processes, the transportation network, the power grid, the internet of things, and many more [1]. Resiliency against attacks is a critical aspect of CPS operation, particularly given the challenges posed by the complex interconnections between the different layers. At the same time, this complexity also presents an opportunity to exploit the features of CPSs to develop approaches to security. This paper investigates the extent to which switching can be leveraged as a defense mechanism to provide protection against attacks in the stabilization of CPSs.

Literature review: CPSs are specially vulnerable because of the unexpected effects resulting from the interaction of the cyber and physical worlds across the communication, computation, and control layers [2], [3]. These include embedded systems prone to bugs and attacks, communication networks under attack, the susceptibility to the flaws of individual components, and increased functionality that opens up new vulnerabilities. Depending on which layer (perception execution, data transmission, or application control) the attacks are injected in, CPSs may suffer from different levels of malfunction and different countermeasures need to be adopted [4], [5], [6], [7], [8]. Unlike classical false data injection attacks, which are normally modeled as additive corruptions to the sensor or actuator signals [9], the type of attacks considered here directly alter the topology of the network or the dynamics of the system. These type of attacks are common for CPSs; for example, the sabotage in the topology can be achieved via denial-of-service (DoS) attack, causing congestion in the communications [10]. The sabotage in the dynamics can be achieved via load altering [11], [12] or pole-dynamics [13] attacks. Whether a CPS is attacked on its topology or dynamics, the attack results in large uncertainties in the system parameters. Hence a valid design approach to preserve its functionality is to synthesize robust controllers against uncertainty in the parameters, for which there is a large body of literature, see e.g. [14] and references therein. It is known [15] that a single, static feedback controller has limited strength against large time-invariant uncertainties. Its viability becomes even more restrictive in the face of time-varying uncertainties. Hence, it becomes necessary to design time-varying – either switched or continuously varying – controllers to ensure the secure operation of CPSs with uncertainties in their parameters, especially when these are caused by time-varying attacks with potentially considerable destabilizing effects.

When a switched defense mechanism is adopted, the CPS then becomes a switched system with uncertainties. The research [16], [17], [18], [19] on the stability of such systems typically considers switching sequences that are independent from the uncertainties of the parameters. Instead, in our study here, both are coupled, as the switching arises as a defensive response to the uncertainty caused by the attack signals. We borrow the idea of partitioning the set of uncertainties from [17], [18], albeit in these works such partition is given a priori, whereas here is the outcome of our design to obtain a finite set of candidate defenses. Recent work [20], [21] on event-triggered control has considered stabilization for systems with uncertain parameters in general, but not specifically driven by adversarial attacks. Finally, we note that gain-scheduling [22], [23] is a popular approach for controlling parameter-varying systems. It proceeds by designing first a collection of candidate controllers for the systems at different operation points and later interpolating them with a continuous time-varying controller. However, the reliance of the controller on knowledge of the time-varying parameters makes it not applicable to our setting, as the uncertainties caused by the attacker are not precisely known by the defender.

This work was partially supported by AFOSR Award FA9550-19-1-0235.

Shenyu Liu is with the School of Automation, Beijing Institute of Technology shenyuliu@bit.edu.cn. Sonia Martínez and Jorge Cortés are with the Department of Aerospace and Mechanical Engineering, University of California, San Diego, {soniamd, cortes}@ucsd.edu. During the preparation of this work, S. Liu was affiliated with UC San Diego.

Statement of contributions: We study the problem of stabilizing a cyber-physical system described by a linear dynamics, where defense and attack actions correspond to structured additive perturbations to the system matrix. The attacker is aware of the employed defense at all times and injects a piecewise Lipschitz time-varying attack signal aiming at destabilizing the system. The defender is unaware of the specific value of the attack and aims to preserve system stability by injecting a switched defense signal. Our first contribution is the design of a divide-and-conquer strategy to partition the set of possible attacks into finitely many subsets such that, for each of them, there exists a single defense that can handle arbitrary attack signals taking values in the subset. We identify conditions on the desired convergence rate of the system such that the proposed algorithm outputs an admissible partition, a set of finite candidate defenses, along with accompanying Lyapunov certificates. Our second contribution synthesizes a defense mechanism that switches among the candidate defenses. Building on the output of the divide-and-conquer strategy, we develop a Lyapunov-based criterion that employs the system state and its time derivative to determine (despite the lack of knowledge of the time-varying attack signal) whether the current defense is stabilizing the system at the desired rate. Motivated by practical considerations, we consider the case where there might be a delay between the determination of a defense switch and its actual implementation. Our third contribution is the characterization of the properties of the resulting switched time-varying system with delay. Specifically, we find an upper bound for the implementation delay under which the corresponding Lyapunov certificate is guaranteed to be decreasing after a switch. We also bound the evolution of the system trajectories under the switched defense mechanism and provide precise conditions on the switching frequency that guarantee global exponential stability of the cyber-physical system with a prescribed convergence rate. Simulations on a network system with a compromised agent and a power system subject to dynamic load attacks provide further validation of the results.

Organization: Section II introduces the problem formulation, including the dynamics for the cyber-physical system, the way in which attack and defense actions affect it, and the information available to the attacker and the defender. Section III proposes an algorithm for finding candidate defenses and analyzes its correctness properties and performance. Section IV builds on the identified set of candidate defenses to design a switched defense mechanism. Our exposition provides conditions on the implementation delay of the defense and the switching frequency to guarantee system stability. Section V provides two examples of cyber-physical systems where the proposed defense mechanism is implemented. We gather our conclusions and ideas for future work in Section VI.

Notation: Let $\mathbb{R}$ and $\mathbb{R}_{\geq 0}$ denote the set of real and nonnegative real numbers, respectively, $\mathbb{R}^n$ the n-dimensional real space and $\mathbb{R}^{n\times n}$ the space of $n \times n$ real matrices. For a vector $x \in \mathbb{R}^n$, $|x|$ denotes its 2-norm and for a matrix $A \in \mathbb{R}^{n\times n}$, $\|A\|$ denotes its induced 2-norm. A matrix $P \in \mathbb{R}^{n\times n}$ is denoted as $P \succ 0$ (resp. $P \succeq 0$, $P \prec 0$, $P \preceq 0$) if it is symmetric and postive definite (resp. postive semi-definite, negative definite, negative semi-definite). We let $\sigma_{\max}(A)$, $\sigma_{\min}(A)$ denote the largest and smallest singular values of A. Note that $\|A\| = \sigma_{\max}(A)$, so we use both notations interchangeably. When $P \succ 0$, then $\sigma_{\max}(P), \sigma_{\min}(P)$ correspond to the largest and smallest eigenvalues of P, respectively. The matrix $A \in \mathbb{R}^{n\times n}$ is Hurwitz if all its eigenvalues are in the left half plane. For $r > 0$ and $x \in \mathbb{R}^n$, $\mathbb{B}^n(x,r) := \{x' \in \mathbb{R}^n : |x' - x| < r\}$ denotes the n-dimensional open ball (we omit the superscript n when the dimension is clear from the context). We let $\otimes$ denote the Kronecker product of matrices and vec the vectorization of matrices.

Consider an n-dimensional time-varying system

$$\dot{x} = f(t,x) \tag{1}$$

with well-defined solutions. We denote by $t \mapsto x(t)$ its solution with initial condition $x(0) = x_0$. The system (1) is *globally exponentially stable* (GES) with convergence rate λ if there exists $c, \lambda > 0$ such that $|x(t)| \leq c|x_0|e^{-\lambda t}$, for all $t \geq 0$ and all $x_0 \in \mathbb{R}^n$.

II. Problem formulation

Consider the following n-dimensional, continuous-time, linear time-varying system

$$\dot{x}(t) = A(\nu(t), \omega(t))x(t), \tag{2a}$$

where the system matrix $A(\nu, \omega) : \mathbb{R}^{m_\nu} \times \mathbb{R}^{m_\omega} \mapsto \mathbb{R}^{n\times n}$ takes the form

$$A(\nu, \omega) := A + \sum_{\alpha=1}^{m_\nu} \nu_\alpha D_\alpha + \sum_{\beta=1}^{m_\omega} \omega_\beta K_\beta. \tag{2b}$$

One can think of $A \in \mathbb{R}^{n\times n}$ as the nominal system matrix upon which defensive actions, described by $\{D_\alpha\}_{\alpha=1}^{m_\nu} \subset \mathbb{R}^{n\times n}$, and attack actions, described by $\{K_\beta\}_{\beta=1}^{m_\omega} \subset \mathbb{R}^{n\times n}$, are superimposed. Throughout the paper, these matrices remain constant. The vectors $\nu := (\nu_1, \cdots, \nu_{m_\nu})^\top \in N \subseteq \mathbb{R}^{m_\nu}$ and $\omega := (\omega_1, \cdots, \omega_{m_\omega})^\top \in \Omega \subseteq \mathbb{R}^{m_\omega}$ are the defense and attack signals describing how the corresponding actions are modulated, with N and Ω the defensive action space and attacking action space, respectively. This formulation (2) encompasses several interesting defense/attack problems, as described.

Additive topological attack and defense: Consider the linear system

$$\dot{x} = Ax. \tag{3}$$

An additive topological attack consists of altering some values in the matrix A so that (3) loses certain desirable property (e.g., stability, connectivity of the underlying graph). Conversely, an additive topological defense consists of altering some values in A so that the system retains said property. Note that the elements in A which are subject to attack or defense may be different. Formally, if $S_D, S_K \subseteq \{1, \cdots, n\}^2$ denote the indices of elements in A that are subject to defense or attack, respectively, then the additive topological attack and defense problem studies the dynamics of

$$\dot{x} = \Big(A + \sum_{(\alpha,\beta)\in S_D} \nu_{\alpha\beta} E_{\alpha\beta} + \sum_{(\alpha,\beta)\in S_K} \omega_{\alpha\beta} E_{\alpha\beta}\Big)x, \tag{4}$$

where $E_{\alpha\beta}$ is the basis matrix whose $\alpha\beta$-th element is 1 and all others 0. The system (4) is a particular case of (2).

System stability via static output feedback: Consider the following linear system with outputs

$$\dot{x} = Ax + B_D u_D + B_K u_K, \tag{5a}$$

$$y_D = C_D x, \tag{5b}$$

$$y_K = C_K x. \tag{5c}$$

Here (u_D, y_D) and (u_K, y_K) are the defender and attacker input-output pairs, respectively. Note that B_D might be different from B_K (meaning the inputs from the defender and attacker can be injected into the system differently) and C_D might be different from C_K (meaning the defender and attacker probe the system in different ways). Both defender and attacker employ static output-feedbacks $u_\# = H_\# y_\#$, $\# \in \{D, K\}$: the defender seeks to stabilize the system (5), whereas the attacker wants to de-stabilize it. In closed-loop form, we have

$$\dot{x} = (A + B_D H_D C_D + B_K H_K C_K)x. \tag{6}$$

Let $\nu := \operatorname{vec}(H_D)$ and $\omega := \operatorname{vec}(H_K)$. Note that $\operatorname{vec}(B_D H_D C_D) = (C_D^\top \otimes B_D)\nu$, and hence $B_D H_D C_D$ is linear in ν. Similarly, $B_K H_K C_K$ is linear in ω. Therefore, there exists matrices $\{D_\alpha\}$, $\{K_\beta\}$, which can be computed from B_D, C_D and B_K, C_K, such that (6) can be written in the form (2).

A. Assumptions on attack and defense

Here we describe the assumptions we make regarding attacking and defensive actions to the system (2), as well as the attacker and defender's knowledge.

On the attacker side: We start with the assumptions on the attack action space and attack signal, followed with a discussion of their reasonableness.

Assumption II.1 (Regularity assumptions on the attack action space and attack signal)**.** *The following conditions hold:*

(i) *Ω is compact and convex;*

(ii) *All attack signals $\omega : \mathbb{R}_{\geq 0} \mapsto \Omega$ are right-continuous and piecewise Lipschitz, with Lipschitz constant $L > 0$. In other words, if $\mathbb{T}^a := \{t_1^a, t_2^a, \cdots\}$ is the set of discontinuities of ω, then $|\omega(t) - \omega(s)| \leq L(t-s)$ for any $t > s > 0$ such that $(s,t) \cap \mathbb{T}^a = \emptyset$;*

(iii) *There is $\tau_0 > 0$ such that the dwell time between two consecutive discontinuities of ω must be no less than τ_0; in other words, $|t - s| \geq \tau_0$ for all $t, s \in \mathbb{T}^a, t \neq s$.*

The first condition in Assumption II.1 captures a number of possibilities, e.g., the attacker has a finite budget to carry out its actions, or the fact that a large-sized attack would trigger the defender to employ some other mechanism, causing the system to fall into protected mode, e.g., [11]. Without loss of generality, we assume $\Omega = [0,1]^{m_\omega}$ in the sequel. This is because, due to compactness of Ω, there exists a hypercube $\Omega_{hc} := \Pi_{\beta=1}^{m_\omega}[a_\beta, b_\beta] \subset \mathbb{R}^{m_\omega}$ such that $\Omega \subset \Omega_{hc}$. If we define $\omega'_\beta := \frac{\omega_\beta - a_\beta}{b_\beta - a_\beta}$ for $\beta = 1, \ldots, m_\omega$, it follows from (2b),

$$\begin{aligned} A(\nu,\omega) &= A + \sum_{\alpha=1}^{\omega_\nu} \nu_\alpha D_\alpha + \sum_{\beta=1}^{m_\omega} \big((b_\beta - a_\beta)\omega'_\beta + a_\beta\big) K_\beta \\ &= \Big(A + \sum_{\beta=1}^{m_\omega} a_\beta K_\beta\Big) + \sum_{\alpha=1}^{\omega_\nu} \nu_\alpha D_\alpha + \sum_{\beta=1}^{m_\omega} \omega'_\beta (b_\beta - a_\beta) K_\beta \\ &=: A'(\nu, \omega'), \end{aligned}$$

with $\omega' := (\omega'_1, \cdots, \omega'_{m_\omega})^\top \in [0,1]^{m_\omega}$, i.e., the system dynamics still has the form (2).

The second condition in Assumption II.1 means that the attack signal can occasionally jump, and in between jumps, it can change continuously, with limits on its speed. In particular, this captures the usual piecewise-constant model for DoS attacks in the literature [24], with attack signals inducing on-off-type changes to the system dynamics and being held constant between changes. The piecewise Lispchitzness between jumps considered here clearly allows for a broader class of attack signals. The last condition in Assumption II.1 imposes a minimal dwell time between consecutive jumps of the attack signal. This can be the result of physical or computational limits on the attacker's capacity. From a practical standpoint, this is reasonable if the attacker requires some non-negligible operational time to physically implement its actions.

Beyond de-stabilizing the dynamics, we do not enter into the specific goals of the attacker or the procedures it uses to decide its attack signal, so long as it satisfies Assumption II.1. Additionally, we consider the following assumptions on the information available to the attacker.

Assumption II.2 (Assumptions on the attacker's knowledge)**.** *The following assumptions hold:*

(i) *The attacker knows the matrices A, D_α, K_β in* (2b)*. (i.e., the attacker knows the functional form of the map $(\nu,\omega) \mapsto A(\nu,\omega)$);*

(ii) *The attacker knows $x(t)$;*

(iii) *The attacker knows the defense signal $\nu(t)$.*

On the defender side: Similarly, we consider the following assumptions for the defender.

Assumption II.3 (Regularity assumptions on the defensive action space and defense signal)**.** *The following conditions hold:*

(i) *N is convex;*

(ii) *There is $\lambda_{\max} \geq 0$ such that, for each $\hat{\omega} \in \Omega$, there exists $\hat{\nu} \in N$ so that $A(\hat{\nu}, \hat{\omega}) + \lambda_{\max} I$ is Hurwitz;*

(iii) *The defense signal $\nu : \mathbb{R}_{\geq 0} \to N$ is right-continuous and piecewise constant, and there exists a finite set $N^f \subset N$ where all defense signals take value.*

The second condition in Assumption II.3 means that, for any attack action $\hat{\omega} \in \Omega$, one can always find a defense action $\hat{\nu} \in N$ such that the rightmost eigenvalue of $A(\hat{\nu}, \hat{\omega})$ is at least $\lambda_{\max}$ away from the imaginary axis. This condition is reasonable because, if it is violated, then there exists $\hat{\omega} \in \Omega$ such that $A(\nu, \hat{\omega})$ is not Hurwitz for all $\nu \in N$. Consequently if the attacker applies the constant attack $\omega(t) = \hat{\omega}$ to the system (2), then there is no static defense action that can stabilize it.

The last condition of Assumption II.3 is motivated by ease-of-implementation and complexity considerations, and means that the defense signal ν only takes finitely many values. The discontinuities of this map are *switches*, which are triggered by some *switch-triggering events* that depend on the defender's knowledge about the system.

Similarly to what we did for the attacker, we summarize the assumptions on the defender's knowledge below.

Assumption II.4 (Assumptions on the defender's knowledge)**.** *The following assumptions hold:*

(i) The defender knows the matrices A, D_α, K_β in (2b)*;*
(ii) The defender knows $x(t), \dot{x}(t)$;
(iii) The defender knows the set Ω, but it does not know the attack signal $\omega(t)$.

Comparing Assumption II.4 with Assumption II.2, we see that the defender has fewer information about the system in the sense that the value of $\omega(t)$ is unknown to the defender. This assumption of unknown attack signals is reasonable when the attacks are stealthy [25], [26], [27]. Assumption II.4 also states that both the state $x(t)$ and $\dot{x}(t)$ are known to the defender, even though the defender does not know $\omega(t)$. Knowledge of $\dot{x}(t)$ can be obtained by a concurrently running state estimator, e.g. [28], or approximated with the information of x over a short time interval $[t, t+\delta]$. As such, we do not require $\dot{x}(t)$ to be known exactly at time t; it can be obtained after a short delay. Further discussion is provided in Section IV-A when introducing the implementation delay.

B. Objectives for the defender

In the scenario described above, the defender aims to accomplish the following two objectives:

O.1 Identify a finite set $N^f \subset N$, such that for each $\hat{\omega} \in \Omega$, there exists $\hat{\nu} \in N^f$ so that the matrix $A(\hat{\nu}, \hat{\omega})$ is Hurwitz;
O.2 Design a switch-triggering mechanism so that the resulting defense signal $\nu(t) : \mathbb{R}_{\geq 0} \mapsto N^f$ makes the system (2) GES.

We refer to the elements in N^f as *candidate defenses*. Objective **O.1** means that, when the system is subject to any static attack $\omega \in \Omega$, at least one of the candidate defenses is capable of stabilizing the system. With regards to Assumption II.3, this objective seeks to narrow down the set of candidate defenses to be finite. In general, the construction of such a finite set is nontrivial. Objective **O.2** corresponds to the defense mechanism design problem. Note that, when the attack changes from being static to continuously time-varying, determining whether a switching defense signal in N^f can stabilize the system is already a challenging question, let alone that ω is allowed to occasionally jump. As formulated, the objective is even more challenging, as the defender does not have knowledge of the attacking signal. Sections III and IV describe our approaches to achieve objectives **O.1** and **O.2**, respectively.

III. Finding a finite set of candidate defenses

In this section, we describe our approach to achieve **O.1**. We first show that a finite set of candidate defenses exists and then design an algorithm to identify it.

A. Finding a finite number of candidate defenses is feasible

The following result establishes that, under Assumption II.3, objective **O.1** is feasible.

Lemma III.1 (Feasibility of **O.1**)**.** *Under Assumption II.3, there exist a positive integer p and subsets $\hat{\Omega}_i \subset \Omega$, actions $\hat{\nu}^i \subset N$ and positive-definite matrices $\hat{P}_i \in \mathbb{R}^{n\times n}$ for all $i = 1, \dots, p$ such that*[1] *$\Omega = \cup_{i=1}^p \hat{\Omega}_i$ and for each i and all $\omega \in \hat{\Omega}_i$,*

$$A(\hat{\nu}^i, \omega)^\top \hat{P}_i + \hat{P}_i A(\hat{\nu}^i, \omega) + 2\lambda_{\max}\hat{P}_i \preceq -\frac{1}{2}I. \quad (7)$$

Proof. From Assumption II.3, we have that for each $\hat{\omega} \in \Omega$, there exists $\hat{\nu} \in N$ and $P = P(\hat{\nu}, \hat{\omega}) \succ 0$ such that

$$(A(\hat{\nu}, \hat{\omega}) + \lambda_{\max}I)^\top P + P(A(\hat{\nu}, \hat{\omega}) + \lambda_{\max}I) = -I. \quad (8)$$

Define

$$c_K := \sqrt{\sum_{\beta=1}^{m_\omega} \|K_\beta\|^2}. \quad (9)$$

Using the Cauchy-Schwarz inequality, we deduce

$$\begin{aligned}\|A(\hat{\nu}, \hat{\omega}) - A(\hat{\nu}, \omega)\| &= \Big\|\sum_{\beta=1}^{m_\omega}(\omega_\beta^* - \omega_\beta)K_\beta\Big\| \\ &\leq \sum_{\beta=1}^{m_\omega} |\omega_\beta^* - \omega_\beta|\|K_\beta\| \leq c_K|\hat{\omega} - \omega|,\end{aligned}$$

for any $\omega \in \Omega$. Thus it follows from (8)

$$\begin{aligned}&(A(\hat{\nu}, \omega) + \lambda_{\max}I)^\top P + P(A(\hat{\nu}, \omega) + \lambda_{\max}I) \\ &\quad = (A(\hat{\nu}, \hat{\omega}) + \lambda_{\max}I)^\top P + P(A(\hat{\nu}, \hat{\omega}) + \lambda_{\max}I) \\ &\qquad + (A(\hat{\nu}, \hat{\omega}) - A(\hat{\nu}, \omega))^\top P + P(A(\hat{\nu}, \hat{\omega}) - A(\hat{\nu}, \omega)) \\ &\quad \preceq -I + 2\|P\|c_K|\hat{\omega} - \omega|.\end{aligned}$$

Therefore, as long as $\omega \in \mathbb{B}(\hat{\omega}, \frac{1}{4\|P\|c_K})$,

$$(A(\hat{\nu}, \omega) + \lambda_{\max}I)^\top P + P(A(\hat{\nu}, \omega) + \lambda_{\max}I) \preceq -\frac{1}{2}I.$$

The result now follows by noting that Ω is a compact set and can be covered by a finite collection of sets of the form $\{\hat{\Omega}_i = \mathbb{B}(\hat{\omega}^i, \frac{1}{4\|\hat{P}_i\|c_K})\}_{i=1}^p$, with each triplet $\hat{\omega}^i \in \Omega$, $\hat{\nu}_i \in N$ and $\hat{P}_i = P(\hat{\nu}^i, \hat{\omega}^i)$ satisfying (7). □

Notice that, although the proof of Lemma III.1 does not explicitly identify the set of candidate defenses, it gives the idea of how to achieve objective **O.1**. That is, one can start by taking a subset $\Omega' \subset \Omega$ and verify if there exists $\nu' \in N$ such that $A(\nu', \omega)$ is Hurwitz for all $\omega \in \Omega'$. If Ω' is sufficiently small, the existence of such ν' is guaranteed by Assumption II.3 and the continuity of eigenvalues with respect to matrix elements. We then take other subsets and repeat this process until the union of all subsets equals Ω. In this way, the associated defensive actions for the subsets are the candidate defenses.

[1] For defense signals, we use superscript i to indicate the i-th defense signal and differentiate from the i-th element in the vector, which is normally indexed with a subscript.

B. Divide and conquer to find set of defensive actions

Here we provide a formal description of the procedure described above to find the finite set of candidate defenses. To do this, the first thing we need to establish is a way of verifying whether a valid defense exists with respect to all perturbations in a given subset of Ω. We next deal with this problem for a subset that is a convex polytope. Consider the system

$$\dot{x} = A(\nu, \omega)x, \quad \omega \in \Omega_{\text{poly}}, \tag{10}$$

where $\nu \in N$ is fixed and $\Omega_{\text{poly}} \subseteq \Omega$ is a convex polytope. Notice that, since $A(\nu, \omega)$ is affine in ω, the system is a *polytopic* system. For such systems, it is known [29, Section 5.1] that GES, even if ω is time-varying, can be ensured by finding a quadratic Lyapunov function common to the collection of all systems of the form (10) with $\omega \in C(\Omega_{\text{poly}})$ (here, $C(\Omega_{\text{poly}})$ denotes the collection of vertices of Ω_{poly}). Therefore, to verify whether all the perturbations in R can be defended by a single defensive action, it is sufficient to look for a quadratic Lyapunov function and solve the following optimization problem

$$\text{(P1)} \qquad \underset{\nu, P, \lambda}{\text{maximize}} \ \lambda \tag{11a}$$

$$\text{subject to}$$

$$\nu \in N \tag{11b}$$

$$P \succ 0 \tag{11c}$$

$$A(\nu, \omega)^\top P + PA(\nu, \omega) + 2\lambda P \leq 0, \ \forall \omega \in C(\Omega_{\text{poly}}). \tag{11d}$$

We denote the solution of problem (P1) by (ν^*, P^*, λ^*). In case $\lambda^* > 0$, then the function

$$V(x) := x^\top P^* x, \tag{12}$$

is a common quadratic Lyapunov function for the polytopic system (10), as for all $\omega \in R$ we have $\dot{V}(x) = x^\top (A(\nu^*, \omega)^\top P^* + P^* A(\nu^*, \omega))x \leq -2\lambda^* x^\top P^* x = -2\lambda^* V(x)$. Therefore, the system is GES and ν^* is a valid candidate defense making $A(\nu^*, \omega)$ Hurwitz for all $\omega \in \Omega_{\text{poly}}$.

Remark III.2 (Nonconvexity of problem (P1)). The optimization problem (P1) is nonconvex since the constraints (11d) are bilinear matrix inequalities (BMI). To solve it, one can employ algorithms based on semi-definite programming with BMI constraints, cf. [30], [31] and references therein. In general, such algorithms do not guarantee the global optimality of the solution, albeit for small-scale problems, they tend to produce accurate results with relatively high efficiency. In our case, as long as $\lambda^* > 0$, locally optimal solutions are acceptable. □

We are now ready to describe formally the procedure to find the finite set of candidate defenses, which essentially consists of recursively partitioning the set of attacking actions $\Omega = [0, 1]^{m_\omega}$ until the optimization problem (P1) is solvable with a positive λ on each subset. We refer to this strategy as "iterative bisection of unit hypercube", cf. Algorithm 1 for a formal description in pseudocode. We next provide an informal description of the rationale behind its steps.

Algorithm 1 Iterative bisection of unit hypercube

```
Input: $A, m_\nu, m_\omega, \{D_\alpha\}_{\alpha=1}^{m_\nu}, \{K_\beta\}_{\beta=1}^{m_\omega}, N, \lambda_{\min}$
Output: $N^f, \mathcal{P}, \mathcal{S}, \bar{\lambda}$
 1: $\mathfrak{R} \leftarrow \{[0,1]^{m_\omega}\}$                                ▷ Initialization
 2: $i \leftarrow 0$
 3: $\bar{\lambda} \leftarrow +\infty$.
 4: while $\mathfrak{R} \neq \emptyset$ do ▷ Stopping criterion; when the checklist is empty
 5:     for each $\Omega_{\text{poly}} = \prod_{\beta=1}^{m_\omega} [a_\beta, b_\beta] \in \mathfrak{R}$ do ▷ Test if a valid defense exists for each subset in the checklist
 6:         $C(\Omega_{\text{poly}}) \leftarrow \{(\omega_1, \cdots, \omega_{m_\omega}) \in \Omega_{\text{poly}} : \omega_\beta = a_\beta$ or $b_\beta$, for all $\beta = 1, \cdots, m_\omega\}$
 7:         Solve (P1). Denote the optimizer $\nu^*, P^*, \lambda^*$
 8:         if $\lambda^* > \lambda_{\min}$ then ▷ Exist; record the solution down and remove the subset from the checklist
 9:             $i = i + 1$,
10:             $(\nu^i, P_i, \Omega_i) \leftarrow (\nu^*, P^*, \Omega_{\text{poly}})$,
11:             $\mathfrak{R} = \mathfrak{R} \backslash \{\Omega_{\text{poly}}\}$
12:             if $\lambda^* < \bar{\lambda}$ then ▷ Update $\bar{\lambda}$ if a smaller value has appeared
13:                 $\bar{\lambda} \leftarrow \lambda^*$
14:             end if
15:         else ▷ Does not exist; replace the subset by its bisection in the checklist
16:             $k \leftarrow \arg\max_{j=1,\cdots,m_\omega} b_j - a_j$
17:             $\Omega_{\text{poly}}^- \leftarrow [a_1, b_1] \times \cdots \times [a_{k-1}, b_{k-1}] \times [a_k, \frac{a_k+b_k}{2}] \times [a_{k+1}, b_{k+1}] \times \cdots \times [a_{m_\omega}, b_{m_\omega}]$
18:             $\Omega_{\text{poly}}^+ \leftarrow [a_1, b_1] \times \cdots \times [a_{k-1}, b_{k-1}] \times [\frac{a_k+b_k}{2}, b_k] \times [a_{k+1}, b_{k+1}] \times \cdots \times [a_{m_\omega}, b_{m_\omega}]$
19:             $\mathfrak{R} \leftarrow (\mathfrak{R} \backslash \{\Omega_{\text{poly}}\}) \cup \{\Omega_{\text{poly}}^-, \Omega_{\text{poly}}^+\}$
20:         end if
21:     end for
22: end while
23: $(N^f, \mathcal{P}, \mathcal{S}) \leftarrow \left(\{\nu^i\}_{i=1}^{|N^f|}, \{P_i\}_{i=1}^{|N^f|}, \{\Omega_i\}_{i=1}^{|N^f|}\right)$
```

Informal description: Algorithm 1 stores in the "checklist" $\mathfrak{R}$ all components of Ω for which no common Lyapunov functions has been found yet. $\mathfrak{R}$ is initialized to be $\{\Omega\}$ (Step 1:), and the algorithm terminates once $\mathfrak{R} = \emptyset$. The algorithm initializes the index $i = 0$ to count the total number of subsets in the checklist for which common Lyapunov functions have been found, and a uniform lower-bound on the decay rate $\bar{\lambda} = \infty$ (Steps 2: and 3:). For each $\Omega_{\text{poly}} \in \mathfrak{R}$, the algorithm determines the set $C(\Omega_{\text{poly}})$ and solves (P1), finding the optimizer ν^*, P^*, λ^* (Steps 6: and 7:). If $\lambda^* > \lambda_{\min}$, then a common Lyapunov function whose decay rate meets the prescribed threshold value $\lambda_{\min}$ on Ω_{poly} has been found, in which case the algorithm increments the index i by 1, stores $\nu^*, P^*, \Omega_{\text{poly}}$ at ν^i, P_i, Ω_i, and removes Ω_{poly} from $\mathfrak{R}$ (Steps 8: to 11:). In addition, if $\lambda^* < \bar{\lambda}$, then the uniform lower-bound on the decay rate is updated to be $\bar{\lambda} = \lambda^*$ (Steps 12: to 14:). On the other hand, if $\lambda^* \leq \lambda_{\min}$, we fail to find a common Lyapunov function over R with a sufficiently large decay rate, and thus we need to refine the partition. To this end, note that all the elements in $\mathfrak{R}$, including the selected Ω_{poly}, are hypercubes. The algorithm then identifies a longest edge of Ω_{poly} and evenly bisects Ω_{poly} into two components $\Omega_{\text{poly}}^-, \Omega_{\text{poly}}^+$ along the dimension where that longest edge lies in (Steps 16: to 18:). The algorithm then replaces the element $\Omega_{\text{poly}} \in \mathfrak{R}$ with $\{\Omega^-_{\text{poly}}, \Omega^+_{\text{poly}}\}$ (Step 19:), and the process is repeated for all the remaining elements in $\mathfrak{R}$, until it becomes empty.

C. Analysis of iterative bisection of unit hypercube algorithm

In this section we characterize the convergence properties of Algorithm 1. The following result shows that it terminates in a finite number of steps and finds a finite set of defensive actions, solving problem **O.1**.

Theorem III.3 (Properties of the iterative bisection of unit hypercube algorithm)**.** *Under Assumption II.3, let $\lambda_{\min} \leq \lambda_{\max}$ and denote by $(N^f, \mathcal{P}, \mathcal{S})$ the output of Algorithm 1. Then, the following hold*

(i) The algorithm terminates in a finite number of steps;

(ii) $\bar{\lambda} > \lambda_{\min}$;

(iii) For any $\omega \in \Omega = [0,1]^{m_\omega}$, there exists $k \in \{1, \cdots, |N^f|\}$ such that, for all $x \in \mathbb{R}^n$,

$$x^\top P_k A(\nu^k, \omega) x \leq -\bar{\lambda} x^\top P_k x. \tag{13}$$

Proof. To prove *(i)*, we resort to Lemma III.1 and take subsets $\hat{\Omega}_i \subset \Omega$, actions $\hat{\nu}^i \in N$ and positive-definite matrices $\hat{P}_i \in \mathbb{R}^{n\times n}$ such that $\Omega = \cup_{i=1}^p \hat{\Omega}_i$ and (7) holds for all $\omega \in \hat{\Omega}_i, i = 1, \cdots, p$. Define

$$\hat{\sigma} := \max_{i=1,\cdots p} \|\hat{P}_i\|.$$

Given $\omega \in \Omega$ arbitrary, there exists $i \in \{1, 2, \cdots, p\}$ such that $\omega \in \hat{\Omega}_i$. Thus, for any $\omega' \in \mathbb{B}(\omega, \frac{1}{8c_K\hat{\sigma}})$, we have

$$\begin{aligned}
&A(\hat{\nu}^i, \omega')^\top \hat{P}_i + \hat{P}_i A(\hat{\nu}^i, \omega') + 2(\frac{1}{8\hat{\sigma}} + \lambda_{\min})\hat{P}_i \\
&\quad \leq A(\hat{\nu}^i, \omega')^\top \hat{P}_i + \hat{P}_i A(\hat{\nu}^i, \omega') + 2(\frac{1}{8\hat{\sigma}} + \lambda_{\max})\hat{P}_i \\
&\quad = (A(\hat{\nu}^i, \omega) + \lambda_{\max} I)^\top \hat{P}_i + \hat{P}_i (A(\hat{\nu}^i, \omega) + \lambda_{\max} I) + \frac{1}{4\hat{\sigma}}\hat{P}_i \\
&\qquad + (A(\hat{\nu}^i, \omega') - A(\hat{\nu}^i, \omega))^\top \hat{P}_i + \hat{P}_i (A(\hat{\nu}^i, \omega') - A(\hat{\nu}^i, \omega)) \\
&\quad \preceq -\frac{1}{2} I + \frac{1}{4\hat{\sigma}} \|\hat{P}_i\| I + 2\|\hat{P}_i\| c_K |\omega' - \omega| I \\
&\quad \preceq -\frac{1}{2} I + \frac{1}{4} I + \frac{1}{4} I = 0,
\end{aligned}$$

where c_K is defined in (9) and the inequality $\|A(\hat{\nu}^i, \omega') - A(\hat{\nu}^i, \omega)\| \leq c_K |\omega' - \omega|$ is derived using the same analysis as in the proof of Lemma III.1. Hence, we conclude that $(\nu, P, \lambda) = (\hat{\nu}^i, \hat{P}_i, \frac{1}{8\hat{\sigma}} + \lambda_{\min})$ is a feasible solution of (P1) in Step 7: if $C(\Omega_{\text{poly}}) \subset \mathbb{B}(\omega, \frac{1}{8c_K\hat{\sigma}})$. Note that the radius of this ball is independent of its center; this means that, once the set R defined in Step 5: is small enough, (P1) will have an optimal solution with $\lambda^* \geq \frac{1}{8\hat{\sigma}} + \lambda_{\min} > \lambda_{\min}$ and consequently Ω_{poly} will be removed from $\mathfrak{R}$. Note that, when the checklist $\mathfrak{R}$ is updated, either an element of $\mathfrak{R}$ is removed (Step 11:) or this element is divided into two smaller ones (Step 19:). Hence, in order to show that $\mathfrak{R}$ will become empty in finitely many steps, it suffices to show that any hypercube Ω_{poly}, when repeatedly divided into two smaller hypercubes by bisection along the dimension where the longest edge lies in, will be contained in a ball of radius $\frac{1}{8c_K\hat{\sigma}}$ in finitely many steps.

To this end, let $d_i = b_i - a_i$ denote the length of the edge of Ω_{poly} along the i-th dimension. Before the bisection, the hypercube Ω_{poly} is contained in a ball of radius $r_{\text{before}} = \frac{1}{2}\sqrt{\sum_{i=1}^{m_\omega} d_i^2}$, where the right-hand side is half of the diagonal distance of Ω_{poly}. Now assume that the longest edge lies in the k-th dimension. We have

$$d_k^2 \geq \frac{1}{m_\omega} \sum_{i=1}^{m_\omega} d_i^2 = \frac{4r^2_{\text{before}}}{m_\omega}.$$

After the division, the length of the edge of either Ω^+_{poly} or Ω^-_{poly} along the k-dimension becomes $\frac{d_k}{2}$, while the lengths of the other edges remain the same. Hence, similarly to the previous analysis, both Ω^+_{poly} and Ω^-_{poly} are contained in a ball of radius

$$\begin{aligned}
r_{\text{after}} &= \frac{1}{2}\sqrt{\left(\frac{d_k}{2}\right)^2 + \sum_{i\neq k} d_i^2} = \frac{1}{2}\sqrt{\sum_{i=1}^{m_\omega} d_i^2 - \frac{3}{4} d_k^2} \\
&\leq \frac{1}{2}\sqrt{4r^2_{\text{before}} - \frac{3}{m_\omega} r^2_{\text{before}}} = \sqrt{1 - \frac{3}{4m_\omega}}\, r_{\text{before}}.
\end{aligned}$$

In other words, the radius of the ball containing the hypercube decreases exponentially at the rate of $\sqrt{1 - \frac{3}{4m_\omega}}$ after each bisection. This implies that the radius of the ball that contains the elements in $\mathfrak{R}$ will be smaller than $\frac{1}{8c_K\hat{\sigma}}$ after finitely many bisections. Hence, we conclude that eventually $\mathfrak{R}$ becomes empty, and Algorithm 1 terminates in a finite number of steps.

Statement *(ii)* follows from the construction of $\bar{\lambda}$ in Step 13:, which corresponds to the smallest value among all solutions λ^* computed in (P1) also satisfying the condition $\lambda^* > \lambda_{\min}$. Finally, statement *(iii)* follows from the fact that $\mathcal{S}$ is a partition of Ω and hence, for any $\omega^* \in \Omega$, there exists $k \in \{1, \cdots, |N^f|\}$ such that $\omega^* \in \Omega_k$. According to Step 7:, ν^k, P^k, λ^* is the optimizer of (P1), with $\Omega_{\text{poly}} = \Omega_k$ and $\lambda^* \geq \bar{\lambda}$ by construction. Thus the inequality (11d) holds for all $\omega \in C(\Omega_k)$. Because Ω_k is the convex hull of $C(\Omega_k)$ and (11d) is convex in ω, the inequality (11d) also holds for all $\omega \in \Omega_k$, including ω^*, which implies (13). □

Remark III.4 (Trade-off between convergence rate of solutions and computational complexity of Algorithm 1). Note that Theorem III.3 requires $\lambda_{\min} \leq \lambda_{\max}$ and ensures that the convergence rate of the solutions of (2), when equipped with the proposed defense mechanism, satisfies $\bar{\lambda} > \lambda_{\min}$. If $\lambda_{\min}$ is chosen negative, then the trajectories may not be convergent but divergent with a bounded divergence rate. In order to ensure stability, one must choose $\lambda_{\min} \in [0, \lambda_{\max}]$, and the larger $\lambda_{\min}$ is, the faster the convergence rate one may obtain. Nevertheless, there is a trade-off between the choice of $\lambda_{\min}$ and the computational effort required by Algorithm 1. In fact, the larger $\lambda_{\min}$ is, the harder it becomes to satisfy the condition $\lambda^* > \lambda_{\min}$ in Step 8 and hence the finer the partition needs to be, resulting in a larger computational effort and a larger number of candidate defenses. Moreover, if the value of $\lambda_{\max}$ is unknown, and as a result, one chooses a value of $\lambda_{\min}$ larger than $\lambda_{\max}$, this would cause Algorithm 1 to fail. If stability is the only concern, one can take $\lambda_{\min} = 0$ for the minimal computational complexity of Algorithm 1. □

IV. Switching defense mechanism

In this section, we propose a switching defense mechanism that solves **O.2**. Our exposition characterizes the performance of the strategy and the impact of the switching frequency on system stability. We also consider the possibility that delays are present between the determination of switch-triggering defense events and their actual execution.

A. Defense mechanism as switched time-varying system with delay

In order to explain the rationale behind our algorithm design, let us first introduce some useful notation. From now on, we assume the total number of candidate defenses is given by Algorithm 1 and hence fixed, and denote $\mathcal{I} := \{1, \cdots, |N^f|\}$. Then $N^f = \{\nu^i : i \in \mathcal{I}\}$ and $\mathcal{P} := \{P_i : i \in \mathcal{I}\}$. As discussed in Section II, based on the knowledge under Assumption II.4, the defender essentially aims to find a right continuous, piecewise constant function $\pi : \mathbb{R}_{\geq 0} \mapsto \mathcal{I}$, such that the defense signal $\nu(t) = \nu^{\pi(t)}$ makes the system (2) GES. For convenience, we define the following quantities

$$\overline{\sigma} := \max_{i \in \mathcal{I}} \sigma_{\max}(P_i), \qquad \underline{\sigma} := \min_{i \in \mathcal{I}} \sigma_{\min}(P_i), \tag{14a}$$

$$\kappa := \max_{i \in \mathcal{I}} \frac{\sigma_{\max}(P_i)}{\sigma_{\min}(P_i)}, \quad \mu := \max_{i,j \in \mathcal{I}} \sigma_{\max}(P_i P_j^{-1}). \tag{14b}$$

We have $1 \leq \mu \leq \kappa \leq \frac{\overline{\sigma}}{\underline{\sigma}}$. In addition,

$$P_i \leq \mu P_j \quad \forall i, j \in \mathcal{I}, \tag{15}$$

and for any $i \in \mathcal{I}$ and $x \in \mathbb{R}^n$,

$$\underline{\sigma}|x|^2 \leq \sigma_{\min}(P_i)|x|^2 \leq x^\top P_i x \leq \sigma_{\max}(P_i)|x|^2 \leq \overline{\sigma}|x|^2. \tag{16}$$

Lastly, for each $k \in \mathcal{I}$, define the function $U_k(x, y) : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}$ by

$$U_k(x, y) := x^\top P_k y. \tag{17}$$

Let $k \in \mathcal{I}$ be the defense used by the defender at a given time. From the definition of U_k, we have

$$\frac{d}{dt} U_k(x(t), x(t)) = 2 U_k(x(t), \dot{x}(t)), \tag{18}$$

i.e., $U_k(x(t), \dot{x}(t))$ encodes the rate of change of the function $U_k(x(t), x(t))$. In view of (16), small values of $U_k(x, x)$ correspond to small values of $|x|$. In order to establish GES, we would like $U_k(x(t), x(t))$ to decrease exponentially. Therefore, by monitoring the value of the functions $U_k(x, x), U_k(x, \dot{x})$, the defender can determine whether the current defense signal is stabilizing the system at a desired rate. The defender needs to update the defense signal when $U_k(x(t), \dot{x}(t))$ is not decreasing fast enough.

However, the actual switches of the defense signal might not be implemented immediately when a need to switch is triggered. Delays might be present due to a variety of factors, including the time taken by 1) the estimation of $\dot{x}$ (see our discussion after Assumption II.4); 2) the computation of the new mode to switch to; and 3) a delay in the execution of the switch. From a technical viewpoint, there is no difference in the analysis of the three types of delays. To deal with delay, we consider $\tau_d > 0$, called *implementation delay*, so that, whenever a need to switch the defense signal is triggered at time t, such switch actually is applied to ν at time $t + \tau_d$. This implementation delay captures the cases 1)-3) above. We remark here that even in the ideal scenario were these cases do not happen, the introduction of an implementation delay provides a minimum dwell time condition on the defense signal that prevents chattering.

Algorithm 2 presents the pseudocode formalizing the intuition for the defense mechanism described above. The design parameter $\eta \in (0, 1)$ specifies the desired rate of decrease of $U_k(x(t), x(t))$. After the initialization with the data $N^f, \mathcal{P}, \bar{\lambda}$ from Algorithm 1, the defense mechanism keeps monitoring $x(t)$ and $\dot{x}(t)$, and computes the values of $U_{\pi(t)}(x(t), x(t))$ and $U_{\pi(t)}(x(t), \dot{x}(t))$. If the current defense does not meet the desired rate of decrease, cf. Step 8:, the algorithm selects a new index in $\mathcal{I}$ according to (20), cf. Step 10:, and then after a delay of τ_d, it updates the defense accordingly, cf. Step 11:. In other words, the condition (19) triggers a need to switch the defense signal at some t, and then the defense will actually implement such a switch according to the mode-to-go rule (20) at time $t + \tau_d$. The binary variable f_{delay} ensures that no more switches are considered while the defender is implementing one during the τ_d seconds that follow the last switching trigger.

Algorithm 2 Switched defense mechanism

```
Input: $\eta \in (0, 1)$, Inputs for Algorithm 1
Output: $\nu(t)$
 1: Run Algorithm 1 and compute $N^f, \mathcal{P}, \bar{\lambda}$    ▷ Initialization
 2: $f_{\text{delay}} \leftarrow 0$ at time 0
 3: Pick $i \in \mathcal{I}$ randomly and $(\pi(0), \nu(0)) \leftarrow (i, \nu^i)$
 4: while the system (2) is running do
 5:     Obtain $x(t), \dot{x}(t)$
 6:     Compute $U_{\pi(t)}(x(t), x(t)), U_{\pi(t)}(x(t), \dot{x}(t))$
 7:     if $f_{\text{delay}} = 0$ then    ▷ Not in delay
 8:         if $x(t) \neq 0$ and    ▷ Not decreasing fast enough
                $U_{\pi(t)}(x(t), \dot{x}(t)) \geq -\eta \bar{\lambda} U_{\pi(t)}(x(t), x(t))$    (19)
            then
 9:             A need to switch is triggered; $f_{\text{delay}} \leftarrow 1$ at time $t$
10:             $\pi(t + \tau_d)$ updates to
                $\arg\min_{i \in \mathcal{I}} \frac{U_i(x(t), \dot{x}(t)) + x(t)^\top P_i \big( \sum_{\alpha=1}^{m_\nu} (\nu^i_\alpha - \nu(t)_\alpha) D_\alpha \big) x(t)}{U_i(x(t), x(t))}$    (20)
11:             $\nu(t + \tau_d) \leftarrow \nu^{\pi(t + \tau_d)}$
12:             $f_{\text{delay}} \leftarrow 0$ at time $t + \tau_d$
13:         end if
14:     end if
15: end while
```

Note that the implementation of the switched defense mechanism in Algorithm 2 does not require direct knowledge of the attack signal ω.

B. The mode-to-go rule and the effect of delay

We analyze the switched defense mechanism introduced above and establish properties that will later be used when studying stability of the closed-loop system (cf. Section IV-C). Specifically, we study here the mode-to-go rule (20) and the effect of implementation delay on the algorithm performance. We start with the observation that if the defender knew the attack signal, then when (19) triggers a need to switch, the defender could simply switch to the mode k such that $\omega(t) \in \Omega_k$. As a consequence of Algorithm 1,

$$\begin{aligned}\frac{d}{dt}U_k(x(t), x(t)) &= 2x(t)^\top P_k A(\nu^k, \omega)x(t)\\ &\leq -2\bar{\lambda}x(t)^\top P_k x(t) = -2\bar{\lambda}U_k(x(t), x(t)).\end{aligned}$$

In other words, $U_k(x(t), x(t))$ would decay exponentially at the rate of $-2\bar{\lambda}$, and mode k would give an effective defense. However, from Assumption II.4, the defender does not have knowledge of the attack signal ω and needs an alternative way to select its defense. The following result guarantees that the mode-to-go rule in (20) provides an effective defense.

Lemma IV.1 (A mode-to-go rule for effective defense)**.** *When a need to switch is triggered at time t, the minimization* (20) *in Step 10: of Algorithm 2 gives a mode-to-go k such that* (13) *holds with $\omega = \omega(t)$, $x = x(t)$.*

Proof. For any $s \in \mathbb{R}_{\geq 0}$ and any $i \in \mathcal{I}$, it holds that

$$\begin{aligned}&x(s)^\top P_i A(\nu^i, \omega(s))x(s)\\ &\quad = x(s)^\top P_i\Big(A(\nu(s), \omega(s)) + \sum_{\alpha=1}^{m_\nu}(\nu^i_\alpha - (\nu(s))_\alpha)D_\alpha\Big)x(s)\\ &\quad = x(s)^\top P_i\dot{x}(s) + x(s)^\top P_i\big(\sum_{\alpha=1}^{m_\nu}(\nu^i_\alpha - (\nu(s))_\alpha)D_\alpha\big)x(s)\\ &\quad = U_i(x(s), \dot{x}(s)) + x(s)^\top P_i\big(\sum_{\alpha=1}^{m_\nu}(\nu^i_\alpha - (\nu(s))_\alpha)D_\alpha\big)x(s).\end{aligned}$$

Hence, when a need to switch π is triggered at time t, the minimization in (20) corresponds to

$$\underset{i\in\mathcal{I}}{\text{minimize}}\,\frac{x(t)^\top P_i A(\nu^i, \omega(t))x(t)}{x(t)^\top P_i x(t)},$$

which, by Theorem III.3 (iii), gives a mode k with an optimal value no larger than $-\bar{\lambda}$. Hence (13) holds with mode k. □

Lemma IV.1 suggests that, by executing Algorithm 2 and if π could be updated instantaneously at time t, the controller would always ensure that the function $U_k(x(\cdot), x(\cdot))$ has the desired decay rate $-2\bar{\lambda}$ after each switch. Nevertheless, because of the implementation delay and the fact that ω is time-varying, $A(\nu^k, \omega(t))$ may have changed with respect to t when ν is actually updated at time $t+\tau_d$, so that the function $U_k(x(\cdot), x(\cdot))$ may no longer decrease at this rate at time $t+\tau_d$. These observations highlight the importance of the next result, which shows that when τ_d is sufficiently small, then $U_k(x(\cdot), x(\cdot))$ is still guaranteed to decrease at a (designer-chosen) fraction of the desired decay rate.

Proposition IV.2 (Bound on the implementation delay to ensure decay rate)**.** *Define*

$$c_A := \max_{\nu\in N^f,\,\omega\in[0,1]^{m_\omega}} \|A(\nu, \omega)\|. \tag{21}$$

Assume that a need to switch is triggered at time s, the attack signal ω is Lipschitz with Lispchitz constant L over the time interval $(s, s+\tau_d]$, and the implementation delay satisfies

$$\tau_d < \frac{(1-\eta)\bar{\lambda}}{(4c_A^2 + c_K L)\kappa^2}, \tag{22}$$

where c_K is defined in (9) *and κ in* (14b)*. Then,*

$$\begin{aligned}&U_{\pi(s+\tau_d)}(x(s+\tau_d), \dot{x}(s+\tau_d))\\ &\qquad\qquad < -\eta\bar{\lambda}U_{\pi(s+\tau_d)}(x(s+\tau_d), x(s+\tau_d)).\end{aligned}$$

Proof. Since ω is Lipschitz over $(s, s+\tau_d]$, it is differentiable almost everywhere over the interval and, whenever the derivative exists, it is upper bounded by L. Therefore, with a constant defense signal $\bar{\nu}$,

$$\begin{aligned}\|\dot{A}(\bar{\nu}, \omega(t))\| &= \|\sum_{\beta=1}^{m_\omega}\dot{\omega}_\beta K_\beta\| \leq \sum_{\beta=1}^{m_\omega}|\dot{\omega}_\beta|\|K_\beta\|\\ &\leq \sqrt{\sum_{\beta=1}^{m_\omega}|\dot{\omega}_\beta|^2\sum_{\beta=1}^{m_\omega}\|K_\beta\|^2} \leq c_K L,\end{aligned} \tag{23}$$

for almost all $t \in (s, s+\tau_d]$. Suppose that π switches from j to $k \in \mathcal{I}$ at time $s+\tau_d$. Denote

$$r_k(t) := \frac{x(t)^\top P_k A(\nu^k, \omega(t))x(t)}{x(t)^\top P_k x(t)}.$$

By definition, $r_k(t)$ is continuous over $[s, s+\tau_d]$ and it follows from Lemma IV.1 that $r_k(s) \leq -\bar{\lambda}$. In addition, since $\dot{x}(t) = A(\nu^j, \omega(t))x(t)$ for $t \in (s, s+\tau_d)$, we have

$$\begin{aligned}\dot{r}_k(t) = (x^\top P_k x)^{-2}\big(&(x^\top A_j^\top P_k A_k x + xP_k A_k A_j x\\ &+ xP_k\dot{A}_k x)x^\top P_k x - 2(x^\top P_k A_k x)(x^\top P_k A_j x)\big),\end{aligned}$$

where we have abbreviated $x(t)$ by x and $A(\nu^\#, \omega(t))$ by $A_\#$ for $\# = j, k$. Taking the norm on both sides and applying the bounds (14) and (23), we conclude

$$\begin{aligned}|\dot{r}_k(t)| &\leq \frac{(4\|A_j\|\|A_k\| + \|\dot{A}_k\|)\sigma_{\max}(P_k)^2|x|^4}{(\sigma_{\min}(P_k)|x|^2)^2}\\ &\leq (4c_A^2 + c_K L)\kappa^2\end{aligned}$$

for all $t \in (s, s+\tau_d)$, which implies that $r_k(t)$ is Lipschitz. Thus, when (22) holds,

$$\begin{aligned}&\frac{U_{\pi(s+\tau_d)}(x(s+\tau_d), \dot{x}(s+\tau_d))}{U_{\pi(s+\tau_d)}(x(s+\tau_d), x(s+\tau_d))} = r_k(s+\tau_d)\\ &\qquad \leq r_k(s) + \tau_d(4c_A^2 + c_K L)\kappa^2 < -\eta\bar{\lambda},\end{aligned}$$

which concludes the proof. □

Recall that the need to switch is determined by the satisfaction of inequality (19), which is ruled out by Proposition IV.2 at time $t+\tau_d$. In other words, when τ_d is sufficiently short and when ω is Lipschitz during the implementation delay, then a need to switch again will not be triggered immediately after the implementation of a switch.

Remark IV.3 (Relation between dwell-time of defense and rate of change of attack). Note from (22) that the bound on τ_d decreases when the Lipschitz constant L of the attack signal increases, and it approaches to 0 as L increases to infinity. This qualitative relationship is expected, reflecting the fact that fast response is expected from the defender in order to catch up with a faster time-varying attack signal. On the other hand, we still need $\tau_d < \frac{(1-\eta)\bar{\lambda}}{4c_A^2\kappa^2} < \infty$ when $L = 0$. In other words, even when the attack signal is static, $U_k(x(t), x(t))$ is still not ensured to decrease exponentially at the desired rate forever and the defense signal might still need to switch. This is because, at any given time t, the defense mechanism reasons myopically about the defense such that the corresponding function $U_k(x(t), x(t))$ has the fastest decay rate at this moment, which does not necessarily imply that such defense remains so for future times $\tau \geq t$ (or even that the function keeps decreasing in the future). This is because the defender does not know whether ω actually belongs to Ω_k, and so the current defense at time t might not be the "correct" defense designed to handle the attack ω. In fact, as discussed above, based only on the information of x and $\dot{x}$, we may never know what is the "correct" defense, so the defense mechanism might need to switch slowly, but indefinitely. Finally, note the dependence of τ_d on the design parameter η: a smaller η means a larger value of τ_d at the cost of a slower convergence rate of the system (2). Hence, the choice of η encodes the trade-off between the tolerance to delay and the convergence rate. □

Now, to examine the combined effects of attacks, defenses, and delays over a long time interval, let $\mathbb{T}^d := \{t_1, t_2, \cdots\}$ be the set of discontinuities of $\nu(t)$, which is also the set of discontinuities of $\pi(t)$. Denote by

$$r(t) = \frac{U_{\pi(t)}(x(t), \dot{x}(t))}{U_{\pi(t)}(x(t), x(t))},$$

which is an evaluation of the temporary decay rate of $U_{\pi(t)}(x(t), x(t))$. By definition, $r(t)$ is piecewise continuous, and the discontinuities are caused by the discontinuities in π or ω. Proposition IV.2 establishes that, if the attack signal is Lipschitz during the delay interval, then after the switch at time t, we have $r(t) \leq -\eta\bar{\lambda}$. However, in the case when ω jumps during the delay interval, it is possible that $r(t) > -\eta\bar{\lambda}$. In this case, according to (19), there is an immediate need for a second switch. This complicates the stability analysis.

To handle this and systematically analyze the stability of the system (2) equipped with our defense mechanism, define by $\tilde{\mathbb{T}}^d = \{\tilde{t}_1, \tilde{t}_2, \cdots\} \subseteq \mathbb{T}^d$ the subset of $\mathbb{T}^d$ containing only those switches such that $r(\tilde{t}_i) \leq -\eta\bar{\lambda}$. We refer to such switches as "effective". The following result characterizes the behavior of the temporary decay rate $r(t)$.

Lemma IV.4 (Behavior of the temporary decay rate)**.** *Assume that* (22) *holds and*

$$\tau_0 > 2\tau_d, \tag{24}$$

where recall that τ_0 is the minimal dwell time between consecutive jumps of the attack signal as in Assumption II.1 (iii), and τ_d is the implementation delay of our proposed switched defense mechanism Algorithm 2. Then when the need to switch π, determined by the satisfaction of (19)*, is initially triggered at time $t^* \in [\tilde{t}_{i-1}, \tilde{t}_i)$, there are only 4 possibilities for the temporary decay rate function $r(t)$ over each time interval $[\tilde{t}_{i-1}, \tilde{t}_i)$, as depicted in Figure 1. Moreover, in the first three cases, $r(t) \leq -\eta\bar{\lambda}$ for all $t \in [\tilde{t}_{i-1}, \tilde{t}_i - \tau_d)$ and $r(t) \leq \kappa c_A$ for all $t \in [\tilde{t}_i - \tau_d, \tilde{t}_i)$ and, in the last case, $r(t) \leq -\eta\bar{\lambda}$ for all $t \in [\tilde{t}_{i-1}, \tilde{t}_i - 2\tau_d)$ and $r(t) \leq \kappa c_A$ for all $t \in [\tilde{t}_i - 2\tau_d, \tilde{t}_i)$.*

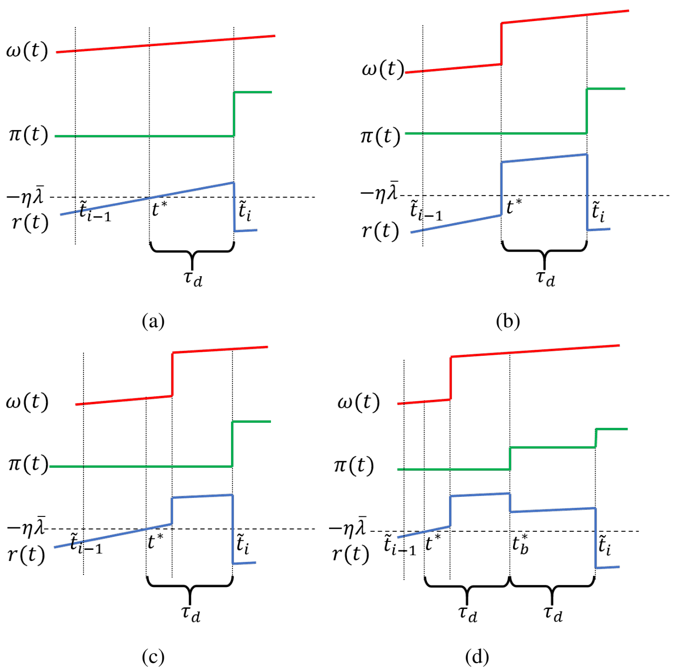


Fig. 1: The plot describes the 4 possibilities that can happen after a switch is triggered during the interval $(\tilde{t}_{i-1}, \tilde{t}_i]$. (a) ω does not jump; (b) ω jumps, triggering a need to switch; (c) ω jumps during the delay for executing a switch, after which $r(\tilde{t}_i) \leq -\eta\bar{\lambda}$; and (d) ω jumps during the delay for executing a switch, after which $r(\tilde{t}_i) > -\eta\bar{\lambda}$ and hence a second switch is needed.

Proof. In case (a) of Figure 1, ω does not jump during the interval. Hence the need to switch π can only be caused by the continuous variation of r. Thus by Proposition IV.2, $\tilde{t}_i = t^* + \tau_d$, yielding an "effective" switch. In all the other cases, ω jumps during the time interval. In case (b), the jump of ω leads to a jump in r, which further creates the need to switch π. Because of (24), ω will not jump again over $(t^*, t^* + \tau_d]$ since it has already jumped at time t^*. Hence again by Proposition IV.2, we have an "effective" switch at $\tilde{t}_i = t^* + \tau_d$. In the remaining two cases, the need to switch is again caused by the continuous variation of r. However, during the implementation delay, ω jumps. Proposition IV.2 is not applicable here and consequently we cannot decide whether this switch, caused by the need at time t^*, is "effective" or not. If $r(t^* + \tau_d) \leq -\eta\bar{\lambda}$, then we are at the situation of case (c) and the switch is "effective". Otherwise, in case (d), there is an immediate need to trigger a next switch at time $t_b^* = t^* + \tau_d$. Note that since ω jumps no earlier than t^*, it again follows from (24) that ω will not jump again over $(t_b^*, t_b^* + \tau_d]$. Hence Proposition IV.2 can be applied over this time interval and conclude that there is an "effective" switch at $\tilde{t}_i = t_b^* + \tau_d$. That concludes all possibilities for the change of $r(t)$ over $[\tilde{t}_{i-1}, \tilde{t}_i)$.

The fact that $r(t) \le -\eta\bar{\lambda}$ for all $t \in [\tilde{t}_{i-1}, \tilde{t}_i - \tau_d)$ in the first three cases, and $r(t) \le -\eta\bar{\lambda}$ for all $t \in [\tilde{t}_{i-1}, \tilde{t}_i - 2\tau_d)$ in the last case can easily be concluded from the plots in Figure 1. Other than those intervals, we can always conclude an upper bound on $r(t)$ directly from its definition:

$$r(t) \le \frac{\sigma_{\max}(P_{\pi(t)})\|A(\nu(t), \omega(t))\|}{\sigma_{\min}(P_{\pi(t)})} \le \kappa c_A,$$

which completes the proof. □

C. Convergence analysis of closed-loop system

Here, we characterize the stability of the system (2) against attacks under the switching defense mechanism. Notice that the system (2) is a time-varying switched system. Since the defender updates its defense signal to stabilize the current mode in which the system is, it is reasonable to conjecture that the system is GES if switches happen slowly. Slow switching is ensured by imposing the following condition

$$\forall t > 0: \quad N_\pi(t) \le \frac{t}{\tau_a} + N_0, \tag{25}$$

for some $\tau_a > 0, N_0 \ge 1$, where $N_\pi(t) := |(0, t] \cap \mathbb{T}^d|$ is the total number of switches of π up to time t. The condition (25) is the classical *average dwell time* (ADT) condition for switched systems [32]. The larger τ_a is, the slower $\pi(t)$ switches. We remark here that according to our defense mechanism, both continuous or discrete changes of $\omega(t)$ may lead to switches in $\pi(t)$. Hence by imposing conditions on the switching frequency of $\pi(t)$, we are indirectly imposing conditions on the switching frequency of $\omega(t)$. We are now ready to present our guarantees on the system stability.

Theorem IV.5 (System stability under the switching defense mechanism and slow switching)**.** *Consider the system* (2)*, where Assumptions II.1, II.2 hold for the attacker and Assumptions II.3, II.4 hold for the defender, subject to the switched defense mechanism Algorithm 2. Assume the implementation delay satisfies* (22) *and* (24)*. Then, the solutions of the system* (2) *satisfy:*

$$|x(t)| \le \sqrt{\frac{\overline{\sigma}}{\underline{\sigma}}} \left(\sqrt{\mu} e^{(\kappa c_A + \eta\bar{\lambda})\tau_d}\right)^{N_\pi(t)} e^{-\eta\bar{\lambda}t}|x_0|. \tag{26}$$

Moreover, if the switching signal π satisfies the ADT condition (25) *with parameter τ_a such that*

$$\tau_a \ge \frac{\frac{1}{2}\ln\mu + (\kappa c_A + \eta\bar{\lambda})\tau_d}{\eta\bar{\lambda} - \lambda}, \tag{27}$$

with $\lambda \in (0, \eta\bar{\lambda})$, then the system (2) *is GES with decay rate λ.*

Proof. Let $V : \mathbb{R}_{\ge 0} \times \mathbb{R}^n \mapsto \mathbb{R}_{\ge 0}$ be defined by $V(t, x) = U_{\pi(t)}(x, x) = x^\top P_{\pi(t)} x$. Note that for almost all $t \in \mathbb{R}_{\ge 0}$, $\frac{d}{dt}V(t, x(t)) = \frac{d}{dt}U_{\pi(t)}(x(t), x(t)) = 2U_{\pi(t)}(x(t), \dot{x}(t)) = 2r(t)U_{\pi(t)}(x(t), x(t)) = 2r(t)V(t, x)$. Hence, under (24), Lemma IV.4 implies that $\frac{d}{dt}V(t, x) \le -2\eta\bar{\lambda}V(t, x)$ for all $t \in [\tilde{t}_{i-1}, \tilde{t}_i - \tau_d)$ in cases (a)-(c) depicted in Figure 1 and all $t \in [\tilde{t}_{i-1}, \tilde{t}_i - 2\tau_d)$ in case (d). Moreover, $\frac{d}{dt}V(t, x) \le 2\kappa c_A V(t, x)$ for all $t \in [\tilde{t}_i - \tau_d, \tilde{t}_i)$ in cases (a)-(c) and all $t \in [\tilde{t}_i - 2\tau_d, \tilde{t}_i)$ in case (d). Therefore, for cases (a)-(c), we have (here, we use t^- to denote the time just before the switch),

$$\begin{aligned} V(\tilde{t}_i, x(\tilde{t}_i)) &\le \mu V(\tilde{t}_i^-, x(\tilde{t}_i^-)) \\ &\le \mu e^{2\kappa c_A \tau_d} V(t^*, x(t^*)) \\ &\le \mu e^{2\kappa c_A \tau_d} e^{-2\eta\bar{\lambda}(t^* - \tilde{t}_{i-1})} V(\tilde{t}_{i-1}, x(\tilde{t}_{i-1})) \\ &= \mu e^{2(\kappa c_A + \eta\bar{\lambda})\tau_d} e^{-2\eta\bar{\lambda}(\tilde{t}_i - \tilde{t}_{i-1})} V(\tilde{t}_{i-1}, x(\tilde{t}_{i-1})), \end{aligned}$$

where we have used the inequality $P_j \le \mu P_i$ for all $i, j \in \mathcal{I}$ (which follows from the definition of μ in (14)) to bound the change of V over any switch. Similarly, case (d), we have

$$\begin{aligned} V(\tilde{t}_i, x(\tilde{t}_i)) &\le \mu V(\tilde{t}_i^-, x(\tilde{t}_i^-)) \\ &\le \mu e^{2\kappa c_A \tau_d} V(t_b^*, x(t_b^*)) \\ &\le \mu^2 e^{2\kappa c_A \tau_d} V((t_b^*)^-, x((t_b^*)^-)) \\ &\le \mu^2 e^{4\kappa c_A \tau_d} V(t^*, x(t^*)) \\ &\le \mu^2 e^{4\kappa c_A \tau_d} e^{-2\eta\bar{\lambda}(t^* - \tilde{t}_{i-1})} V(\tilde{t}_{i-1}, x(\tilde{t}_{i-1})) \\ &= \left(\mu e^{2(\kappa c_A + \eta\bar{\lambda})\tau_d}\right)^2 e^{-2\eta\bar{\lambda}(\tilde{t}_i - \tilde{t}_{i-1})} V(\tilde{t}_{i-1}, x(\tilde{t}_{i-1})). \end{aligned}$$

Now note that in each of the cases (a)-(c), one switch of π occurs over the interval $(\tilde{t}_{i-1}, \tilde{t}_i]$ (in fact, exactly at $\tilde{t}_i$) and, in case (d), two switches of π occur over the interval $(\tilde{t}_{i-1}, \tilde{t}_i]$. Hence with the convention $\tilde{t}_0 = 0$ and by concatenating the changes from $V(\tilde{t}_{i-1}, x(\tilde{t}_{i-1}))$ to $V(\tilde{t}_i, x(\tilde{t}_i))$, for $i = 1, 2, \cdots$ together, we conclude

$$V(t, x(t)) \le \left(\mu e^{2(\kappa c_A + \eta\bar{\lambda})\tau_d}\right)^{N_\pi(t)} e^{-2\eta\bar{\lambda}t} V(0, x_0),$$

where recall that $N_\pi(t)$ denotes the total number of switches of π over the interval $(0, t]$. Therefore, it follows from (16) that

$$\begin{aligned} \underline{\sigma}|x(t)|^2 &\le V(t, x(t)) \\ &\le \left(\mu e^{2(\kappa c_A + \eta\bar{\lambda})\tau_d}\right)^{N_\pi(t)} e^{-2\eta\bar{\lambda}t} V(0, x_0) \\ &\le \left(\mu e^{2(\kappa c_A + \eta\bar{\lambda})\tau_d}\right)^{N_\pi(t)} e^{-2\eta\bar{\lambda}t} \overline{\sigma}|x_0|^2, \end{aligned}$$

yielding (26). To show that the system is GES when π satisfies an ADT condition with the inequality (27), notice that in this case

$$\begin{aligned} &\Big(\frac{1}{2}\ln\mu + (\kappa c_A + \eta\bar{\lambda})\tau_d\Big) N_\pi(t) \\ &\qquad \le (\eta\bar{\lambda} - \lambda)t + N_0\Big(\frac{1}{2}\ln\mu + (\kappa c_A + \eta\bar{\lambda})\tau_d\Big). \end{aligned}$$

Taking the exponential on both sides and substituting into (26), we conclude $|x(t)| \le c|x_0|e^{-\lambda t}$ with $c := \sqrt{\frac{\overline{\sigma}}{\underline{\sigma}}} e^{N_0\left(\frac{1}{2}\ln\mu + (\kappa c_A + \eta\bar{\lambda})\tau_d\right)}$, concluding the result. □

Remark IV.6 (Qualitative interpretation of Theorem IV.5). Note that the condition in Theorem IV.5 is conservative, given that various approximations are made in order to get the constants κ, c_A, μ. As a result, the right-hand side of (27) might be overestimated. In some cases, one can easily check that this condition is satisfied. For instance, if Algorithm 1 gives a common Lyapunov function (i.e., all P_i's are the same) and if the defense mechanisms has no implementation delay, we have $\mu = 1$ and $\tau_d = 0$, and therefore the right-hand side of (27) is 0. In this case, the defense mechanism is guaranteed to stabilize the system no matter how fast the attack signal varies. Generalizing this observation qualitatively, one can interpret the condition as saying that GES of the system (2) is guaranteed when the attack signal varies slowly enough (leading to a slow switching π and hence larger τ_a), the computed P_i's are similar to each other (small μ), and the delay in the defense mechanism is short (small τ_d). □

V. Simulations

This section illustrates in two examples how the proposed switched defense mechanism guarantees stability of the attacked system. The first example consists of a network system with a compromised agent and the second one deals with dynamic load altering attacks on a power system.

A. Network with a compromised agent

Consider a network of 5 first-order agents, cf. Figure 2, whose nominal dynamics is given by $\dot{x} = Ax$, with

$$A = \begin{bmatrix} -1 & 0 & 0 & 2 & -2 \\ 0 & -1 & 0 & 2 & -2 \\ 0 & 0 & -1 & 2 & -2 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -3 & 2 \end{bmatrix}.$$

Agents 1, 2, 3 are identical receivers which update their state using the information received from the other two agents. Agent 4 is a defensive influencer which can uniformly change the impact on its propagating information, labeled as ν in Figure 2. Agent 5 is a compromised influencer, which can similarly change the impact on its propagating information. In addition, the attacker identifies the defensive agent in the network, so the compromised agent can also alter the information received from Agent 4. These two types of modifications from the attacker are labeled by ω_1, ω_2 in Figure 2. Consequently, the resulting network system can be formulated in the form of (2),

$$\dot{x} = \big(A + \nu D_1 + \omega_1 K_1 + \omega_2 K_2\big)x, \tag{28}$$

where $D_1, K_1, K_2 \in \mathbb{R}^{5\times 5}$ are matrices such that D_1 has -1 for all the elements in the 4-th column and 0 for all other elements, K_1 has 1 for all the elements in the 5-th column and 0 for all other elements, and K_2 has 1 for the $(5,4)$-th element and all other elements 0. We take $N = \mathbb{R}$ and $\Omega = [0,1]^2$. One can verify that the second condition in Assumption II.3 holds. We then run Algorithm 1 with $\lambda_{\min} = 0$, which results in a partition $\mathcal{S}$ of Ω into the 10 subsets shown in Figure 3 and a convergence rate guarantee of $\bar{\lambda} = 0.081$.

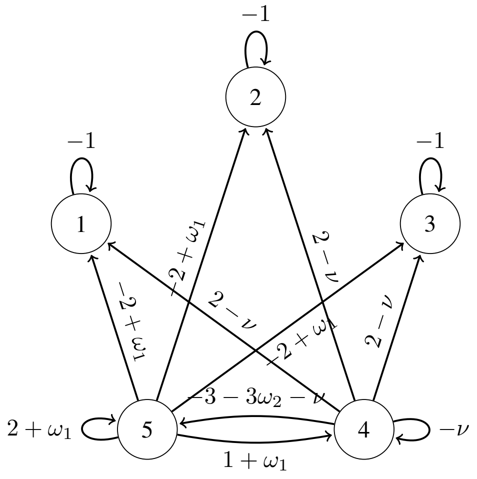


Fig. 2: The network of 5 agents

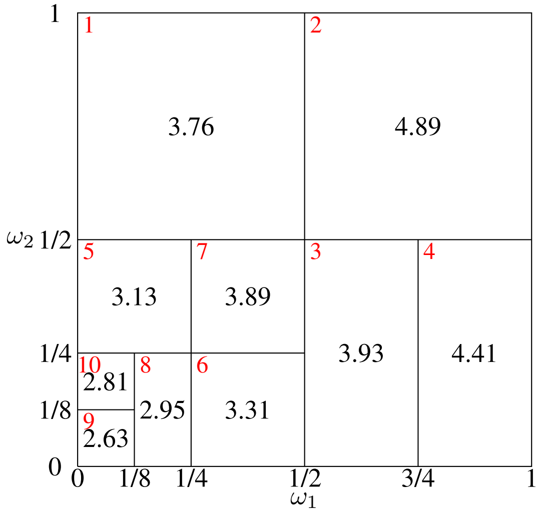


Fig. 3: Partition of Ω generated by Algorithm 1. For each subset, the red number at the corner denotes the index and the black number in the middle denotes the corresponding candidate defense ν.

Defense against pre-set attacks: We first consider the attack signals depicted in the top plot of Figure 4 and examine the system performance under the switched defense mechanism. The attack signals are such that ω_1 is a periodic ramp function with slope $1/12$ and period 12. ω_2 is a binary signal that starts at 0, jumps to 1 at time 8 and switches back to 0 at time 12. The attack signal $\omega = [\omega_1\ \omega_2]^\top$ is piecewise Lipschitz and Assumption II.1 is satisfied with $L = 1/12, \tau_0 = 4$. The other two plots in Figure 4 illustrate the state trajectories and switching defense signal evolving under the action of the defense mechanism for the initial state $x(0) = \begin{bmatrix} -1 & 0 & 1 & 1 & 1 \end{bmatrix}^\top$ with parameters $\eta = 0.9$ and $\tau_d = 0.5$. All states converge to 0 under the switching defense mechanism.

Defense against strategic attacks: Next, we consider a strategic attack and examine the system performance under the switched defense mechanism. For each $\nu \in N^f$, let $\omega^\nu \in \Omega$ be the "worst-case" attack, in the sense that the rightmost eigenvalue of $A(\nu, \omega^\nu)$ has the largest real part (this can be obtained using the structured pseudospectral abscissa, cf. [33]). We then define the attack signal $w(t)$ such that $w(0) = 0$ and

$$\dot{\omega}(t) := \begin{cases} L\frac{\omega^{\nu(t)} - \omega(t)}{|\omega^{\nu(t)} - \omega(t)|} & \text{if } \omega(t) \neq \omega^{\nu(t)}, \\ 0 & \text{if } \omega(t) = \omega^{\nu(t)}. \end{cases} \tag{29}$$

Intuitively, the dynamics (29) is such that, when $\omega(t)$ is not the "worst-case" attack with respect to the current defense signal

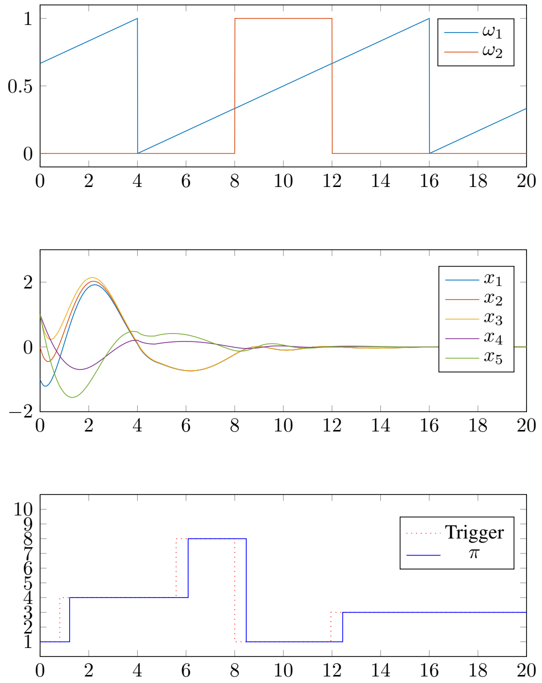


Fig. 4: From top to bottom: attack signals, state trajectories, and switching defense signal of system (28). The red dotted line in the bottom plot indicates the triggers for switches, which is always τ_d ahead of π.

$\nu(t)$, then it changes linearly with rate L towards this "worst-case" value. If $\omega(t)$ is already the "worst-case" attack, then it does not change. With this definition, the attack satisfies Assumption II.1 for the Lipschitz constant L and any $\tau_0 > 0$. Moreover, since it can be computed that for any $\nu \in N$, there exists $\omega \in \Omega$ such that $A(\nu, \omega)$ is unstable, such an attack signal can destabilize the system if no defense action, or a defense action which eventually becomes steady is taken.

We study the performance of the defense strategy for different values of the Lipschitz constant L of the attacker and the implementation delay τ_d of the defense mechanism. We use $\eta = 0.8$ and, for each pair of L, τ_d, we simulate 100 trials with initial state obeying a normal distribution $N(0, 1)$. Tables I, II, and III show, respectively, the average relative overshoot, defined as $\max_{t \geq 0} |x(t)|/|x(0)|$, the average settling time, defined as $\min\{t \geq 0 : |x(s)| \leq 0.02|x(0)| \ \forall s \geq t\}$, and the average number of switches up to time $t = 100$. One can observe the tendency[2] of the relative overshoot and the settling time increase with τ_d, and the number of switches increases with τ_d and L. When $\tau_d \geq 0.4$, the settling time increases significantly when L increases. Moreover, when $\tau_d = 0.5$ and $L \geq 1.5$, the system becomes unstable. This is expected, since when the delay becomes large, the defense action cannot catch up with the attack action and hence the destabilizing effect brought by the attack action will accumulate.

We also observe that, although Algorithm 1 produces 10 candidate defenses, some of them are never used in the simulations. This indicates that, depending on the state trajectory, some modes might always be dominated by other modes while determining the "mode-to-switch" in Step 10: of the defense mechanism. This suggests the possibility of refinements to the defense mechanism that prune dominated modes to improve its efficiency.

L / τ_d	0.5	1	1.5	2	2.5
0.1	3.54	3.79	3.94	3.59	3.51
0.2	3.98	4.02	3.95	3.98	3.64
0.3	3.66	3.88	4.42	3.45	3.64
0.4	3.97	4.25	4.35	3.86	4.09
0.5	3.74	4.48	unstable	unstable	unstable

TABLE I: Relative overshoot.

L / τ_d	0.5	1	1.5	2	2.5
0.1	9.8	10.3	9.2	8.8	8
0.2	11.2	9	7.7	6.8	7.5
0.3	11	10.5	11.2	10.2	9.2
0.4	13.4	16	37.4	40.6	42.8
0.5	16.2	51.1	unstable	unstable	unstable

TABLE II: Settling time.

L / τ_d	0.5	1	1.5	2	2.5
0.1	56	57	64	91	75
0.2	43	77	78	56	83
0.3	51	66	90	90	92
0.4	55	78	87	93	95
0.5	60	74	82	87	91

TABLE III: Number of switches over $[0, 100]$.

B. Power system subject to dynamic load altering attacks

Here we consider an example taken from [11] of a power system under dynamic load altering attacks. The system is composed of n^G generator buses and n^L load buses. At each generator bus, a linear swing equation models the generator dynamics. A proportional-integral controller determines the mechanical power input for each generator. The overall linear state-space descriptor system can be written as

$$\begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & -M & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\delta} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{\varphi} \end{bmatrix} = \tag{30}$$

$$\begin{bmatrix} 0 & 0 & I & 0 \\ 0 & 0 & 0 & -I \\ K^I + H^{GG} & H^{GL} & K^P + D^G & 0 \\ H^{LG} & H^{LL} & 0 & D^L \end{bmatrix} \begin{bmatrix} \delta \\ \theta \\ \psi \\ \varphi \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ P^L \end{bmatrix},$$

where $\delta, \psi \in \mathbb{R}^{n^G}$ are the vectors of voltage phase angles and of rotor angular frequency deviations at generator buses, respectively, and $\theta, \varphi, P^L \in \mathbb{R}^{n^L}$ are the vectors of voltage phase angles, of frequency deviations, and of power consumption at load buses, respectively. In addition, M, $D^G \in$

[2] The lack of monotonicity in this tendency can be explained by noting that the specific strategic attack resulting from (29) does not necessarily provide the worst-case attack to the stability of the system. Our stability analysis under the proposed defense mechanism instead provides a worst-case guarantee, so we cannot expect to observe a monotonic behavior with respect to τ_d and L as reflected by Remark IV.3 and Theorem IV.5.

$\mathbb{R}^{n^G \times n^G}$, $D^L \in \mathbb{R}^{n^L \times n^L}$ are diagonal matrices with entries equal to the inertia, damping coefficients of the generators, and damping coefficients of the loads, respectively. Similarly, $K^I, K^P \in \mathbb{R}^{n^G \times n^G}$ are diagonal matrices with entries equal to the integral and proportional controller coefficients of the generators. Lastly,

$$H_{\text{bus}} := \begin{bmatrix} H^{GG} & H^{GL} \\ H^{LG} & H^{LL} \end{bmatrix} \in \mathbb{R}^{(n^G+n^L)\times(n^G+n^L)}$$

is the imaginary part of the admittance matrix.

Defense/attack formulation as a linear time-varying system: Following [11], a dynamic load altering attack on load bus k by feedback from generator bus l is an additional power consumption on the load bus k in the form $P^A_{k,l} = \omega_{k,l}\psi_l$, for some $\omega_{k,l} \in \Omega_{k,l} \subset \mathbb{R}$. Similarly, we can define a dynamic load altering defense on load bus k by feedback from generator bus l to be $P^D_{k,l} = \nu_{k,l}\psi_l$, for some $\nu_{k,l} \in N_{k,l} \subset \mathbb{R}$. If $\mathcal{L}^A, \mathcal{L}^D \subseteq \{1, \cdots, n^L\} \times \{1, \cdots, n^G\}$ denote the sets of attack and defense load-bus pairs, respectively, we have

$$P^L_k = P^S_k + \sum_{l:(k,l)\in\mathcal{L}^A} P^A_{k,l} + \sum_{l:(k,l)\in\mathcal{L}^D} P^D_{k,l} \tag{31}$$

where P^S_k is the constant secured power consumption, which can be ignored during the analysis of stability of the equilibrium. It follows from (31) that if $(k,l) \notin \mathcal{L}^A$ (resp. $(k,l) \notin \mathcal{L}^D$) for all $l \in \{1, \cdots, n^G\}$, then the load bus k is not subject to any load altering attack (resp., defense).

Note that (30) is a differential-algebraic equation, which can be simplified by expressing φ in terms of δ, θ, ψ. Moreover, by replacing P^L with the expression (31), where $P^A_{k,l}, P^D_{k,l}$ are further replaced by their definitions, we can write

$$\begin{bmatrix} \dot\delta \\ \dot\theta \\ \dot\psi \end{bmatrix} = \left(A + \sum_{(k,l)\in\mathcal{L}^A} \omega_{k,l} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & (D^L)^{-1}E_{k,l} \\ 0 & 0 & 0 \end{bmatrix} \right.$$
$$\left. + \sum_{(k,l)\in\mathcal{L}^D} \nu_{k,l} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & (D^L)^{-1}E_{k,l} \\ 0 & 0 & 0 \end{bmatrix} \right) \begin{bmatrix} \delta \\ \theta \\ \psi \end{bmatrix}, \tag{32}$$

where

$$A := \begin{bmatrix} I & 0 & 0 \\ 0 & D^L & 0 \\ 0 & 0 & -M \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 & I \\ H^{LG} & H^{LL} & 0 \\ K^I + H^{GG} & H^{GL} & K^P + D^G \end{bmatrix}$$

and $E_{k,l} \in \mathbb{R}^{n^L \times n^G}$ is the basis matrix whose (k,l)-th element is 1 and all others 0. Hence, we recover the model (2), with ω and ν being the vectors consisting of all the feedback gains $\omega_{k,l}$ and $\nu_{k,l}$, respectively. We assume such gains can be tuned online, i.e., might be time-varying.

Finite set of candidate defenses: For simplicity of presentation, we consider a 5-bus system with 2 generator buses and 3 load buses. Assume that the inertia of each generator is 10 and the damping coefficients is 1 for any generator or load. The imaginary part of the admittance matrix is

$$H_{\text{bus}} = \begin{bmatrix} -5.5 & 2.5 & 2 & 0 & 0 \\ 2.5 & -11.5 & 4 & 0 & 5 \\ 2 & 4 & -14 & 8 & 0 \\ 0 & 0 & 8 & -10 & 2 \\ 0 & 5 & 0 & 2 & -7.8 \end{bmatrix}.$$

Pick $K^I = K^P = 10I$, so that the nominal system matrix A is Hurwitz. Now consider dynamic load altering attacks on load buses 1 and 2 by feedback only from generator bus 1, and assume they saturate at magnitude of 100. In other words, $\mathcal{L}^A = \{(1,1),(2,1)\}$ and $\Omega := \Omega_{1,1} \times \Omega_{2,1} = [-100, 100]^2$. We further assume that all load buses are subject to dynamic load altering defenses with respect to any generator buses and there are no constraints on the feedback gain. In other words, $\mathcal{L}^D = \{1,2,3\} \times \{1,2\}$ and $N := \prod_{(k,l)\in\mathcal{L}^D} N_{k,l} = \mathbb{R}^6$. Using the parameter $\lambda_{\min} = 0.05$ for Algorithm 1, we obtain a partition of Ω into 4 subsets. Table IV lists the corresponding candidate defense signals ν^i and decay rate $\bar\lambda$.

Defense against strategic attacks: We consider the same strategic attack policy proposed in the previous example. Figure 5 illustrates the attack signals, system trajectories, and switching defense signal evolving under the action of the defense mechanism with parameters $\eta = 0.9, L = 20, \tau_d = 1$. We have also simulated 100 trajectories with the same parameters and initial states uniformly randomly taken in $[-1, 1]^7$, and computed the average relative overshoot to be 3.35 and the average settling time to be 40, respectively.

Mode	1	2	3	4	–
ν^i	$\begin{bmatrix} 142.85 \\ 109.93 \\ 12.47 \\ -43.28 \\ -21.13 \\ -4.87 \end{bmatrix}$	$\begin{bmatrix} 142.85 \\ 109.93 \\ -87.53 \\ -43.28 \\ -21.13 \\ -4.87 \end{bmatrix}$	$\begin{bmatrix} 42.85 \\ 109.93 \\ 12.47 \\ -43.28 \\ -21.13 \\ -4.87 \end{bmatrix}$	$\begin{bmatrix} 42.85 \\ 109.93 \\ -87.53 \\ -43.28 \\ -21.13 \\ -4.87 \end{bmatrix}$	$\begin{bmatrix} 240.34 \\ 381.07 \\ -46.39 \\ -199.61 \\ -127.43 \\ -50.19 \end{bmatrix}$
$\lvert\nu^i\rvert$	187.06	206.15	128.1460	154.69	513.5
λ	0.0698				0.0423

TABLE IV: Defense designs for the power system obtained by Algorithm 1.

We remark here that using the parameter $\lambda_{\min} = 0$ for Algorithm 1 gives a single defense which can stabilize the power system for any attack (the defense signal and its corresponding decay rate $\bar\lambda$ are shown in the last column of Table IV). With the same initial condition as for generating the solution in Figure 6, we get different state trajectories, plotted in Figure 5. The overshoot is much larger and it takes longer time for settling. Meanwhile, 100 trajectories under the single defense mechanism with initial states uniformly randomly taken in $[-1, 1]^7$ are simulated, which yield an average relative overshoot of 5.75 and average settling time of 115. This means that in the comparison with the switched defense mechanism, the performance of the single defense fares worse, as the relative overshoot is about 1.7 times larger and the settling time is almost 3 times larger. We have observed a similar relative performance when generating solutions using other values of L and τ_d. Moreover, Table IV also shows that the switched defense requires a significantly smaller control effort ($|\nu|$ of the defense) than the single defense.

VI. Conclusions

We have proposed a switched defense mechanism to globally exponentially stabilize a cyber-physical system subject to attack. The system is described by a linear matrix whose entries can be modified by the attack signal, which is unknown to the defender and has a bounded rate of change. The defense

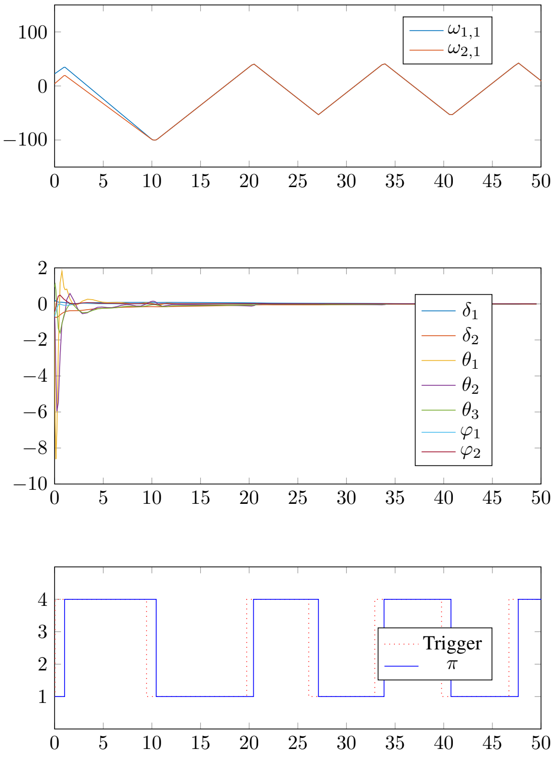


Fig. 5: From top to bottom: attack signals, state trajectories, and switching defense signal of the power system (32).

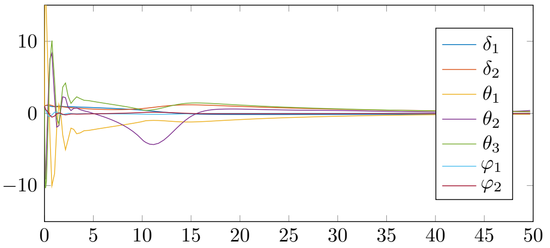


Fig. 6: State trajectories of the power system (32) under a constant defense. Note that the scale of the magnitude axis is different from the one in the middle plot of Figure 5.

mechanism relies on switch-triggering events to modify the system dynamics using information about the state and its derivative. To design it, we have built on a partition of the set of attacks that provides the defender with a finite set of effective candidate defenses. Our stability analysis of the resulting time-varying system incorporates the possibility of an implementation delay in the execution of the defense switches and shows that, as long as the delay is sufficiently short, switches will not be constantly triggered. We have also identified conditions on the switching frequency, under which GES is guaranteed for all allowable attack signals. Simulations on a compromised network system and a power system subject to dynamic load altering attacks illustrate the performance of the proposed defense mechanism. Future work will explore the extension of our results when estimation errors are present and to more general attack signals; refinements to the proposed defense mechanism design, including different partition techniques, alternative optimization problems with LMI constraints, pruning dominated candidate defenses, and distributed implementations; and the consideration of other performance metrics (beyond the convergence rate) for the design of defense mechanisms.

References

[1] S. Roy and S. K. Das, *Principles of Cyber-Physical Systems: An Interdisciplinary Approach*. Cambridge University Press, 2020.

[2] K. D. Kim and P. R. Kumar, "Cyber-physical systems: A perspective at the centennial," *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1287–1308, 2012.

[3] J. Sztipanovits, X. Koutsoukos, G. Karsai, N. Kottenstette, P. Antsaklis, V. Gupta, B. Goodwine, J. Baras, and S. Wang, "Toward a science of cyber-physical system integration," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 29–44, 2012.

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

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

[6] P. Griffioen, S. Weerakkody, B. Sinopoli, O. Ozel, and Y. Mo, "A tutorial on detecting security attacks on cyber-physical systems," in *European Control Conference*, Naples, Italy, 2019, pp. 979–984.

[7] R. Romagnoli, B. H. Krogh, and B. Sinopoli, "Design of software rejuvenation for cps security using invariant sets," in *2019 American Control Conference (ACC)*, 2019, pp. 3740–3745.

[8] L. Cao, X. Jiang, Y. Zhao, S. Wang, D. You, and X. Xu, "A survey of network attacks on cyber-physical systems," *IEEE Access*, vol. 8, pp. 44 219–44 227, 2020.

[9] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, New York, NY, USA, 2009, p. 21–32.

[10] S. Amin, A. A. Cárdenas, and S. S. Sastry, "Safe and secure networked control systems under denial-of-service attacks," *Hybrid systems: Computation and Control*, pp. 31–45, 2009.

[11] S. Amini, F. Pasqualetti, and H. Mohsenian-Rad, "Dynamic load altering attacks against power system stability: Attack models and protection schemes," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 2862–2872, 2018.

[12] H. E. Brown and C. L. Demarco, "Risk of cyber-physical attack via load with emulated inertia control," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 5854–5866, 2018.

[13] H. Jeon and Y. Eun, "A stealthy sensor attack for uncertain cyber-physical systems," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6345–6352, 2019.

[14] I. R. Petersen and R. Tempo, "Robust control of uncertain systems: Classical results and recent developments," *Automatica*, vol. 50, no. 5, pp. 1315–1335, 2014.

[15] F. Wu, X. H. Yang, A. Packard, and G. Becker, "Induced l_2-norm control for LPV systems with bounded parameter variation rates," *International Journal on Robust and Nonlinear Control*, vol. 6, no. 9-10, pp. 983–998, 1996.

[16] G. Zhai, H. Lin, and P. J. Antsaklis, "Quadratic stabilizability of switched linear systems with polytopic uncertainties," *International Journal of Control*, vol. 76, no. 7, pp. 747–753, 2003.

[17] L. I. Allerhand and U. Shaked, "Robust stability and stabilization of linear switched systems with dwell time," *IEEE Transactions on Automatic Control*, vol. 56, no. 2, pp. 381–386, 2011.

[18] ——, "Robust control of linear systems via switching," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 506–512, 2013.

[19] W. Xiang, "Necessary and sufficient condition for stability of switched uncertain linear systems under dwell-time constraint," *IEEE Transactions on Automatic Control*, vol. 61, no. 11, pp. 3619–3624, 2016.

[20] L. Xing, C. Wen, Z. Liu, H. Su, and J. Cai, "Event-triggered adaptive control for a class of uncertain nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 2071–2076, 2017.

[21] Y. Huang and Y. Liu, "Switching event-triggered control for a class of uncertain nonlinear systems," *Automatica*, vol. 108, p. 108471, 2019.

[22] D. J. Leith and W. E. Leithead, "Survey of gain-scheduling analysis and design," *International Journal of Control*, vol. 73, no. 11, pp. 1001–1025, 2000.

[23] P. Apkarian and R. J. Adams, "Advanced gain-scheduling techniques for uncertain systems," in *Advances in Linear Matrix Inequality Methods in Control*, ser. Advances in Design and Control. Philadelphia, PA: SIAM, 2000, pp. 209–228.

[24] C. De Persis and P. Tesi, "Input-to-state stabilizing control under denial-of-service," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 2930–2944, 2015.

[25] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "Revealing stealthy attacks in control systems," in *Allerton Conf. on Communications, Control and Computing*, Monticello, IL, 2012, pp. 1806–1813.

[26] A. O. de Sá, L. F. R. C. Carmo, and R. C. S. Machado, "Covert attacks in cyber-physical control systems," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 1641–1651, 2017.

[27] Y. Mao, H. Jafarnejadsani, P. Zhao, E. Akyol, and N. Hovakimyan, "Novel stealthy attack and defense strategies for networked control systems," *IEEE Transactions on Automatic Control*, vol. 65, no. 9, pp. 3847–3862, 2020.

[28] X. Zhao, H. Liu, J. Zhang, and H. Li, "Multiple-mode observer design for a class of switched linear systems," *IEEE Transactions on Automation Sciences and Engineering*, vol. 12, no. 1, pp. 272–280, 2015.

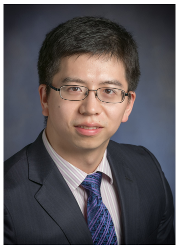
[29] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, ser. Studies in Applied Mathematics. Philadelphia, Pennsylvania: SIAM, 1994, vol. 15.

[30] Q. Tran Dinh, S. Gumussoy, W. Michiels, and M. Diehl, "Combining convex–concave decompositions and linearization approaches for solving BMIs, with application to static output feedback," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1377–1390, 2012.

[31] J. Fiala, M. Kočvara, and M. Stingl, "Penlab: A matlab solver for nonlinear semidefinite optimization," *arXiv preprint arxiv:1311.5240*, 2013.

[32] J. P. Hespanha and A. S. Morse, "Stability of switched systems with average dwell-time," in *IEEE Conf. on Decision and Control*, Shangai, China, Dec. 1999, pp. 2655–2660.

[33] S. Liu, S. Martínez, and J. Cortés, "Iterative algorithms for assessing network resilience against structured perturbations," *IEEE Transactions on Control of Network Systems*, 2022, to appear.



Shenyu Liu (S'16-M'20) received his B. Eng. degree in Mechanical Engineering and B.S. degree in Mathematics from the University of Singapore, Singapore, in 2014. He then received his M.S. degree in Mechanical Engineering from the University of Illinois, Urbana-Champaign in 2015, where he also received his Ph.D. degree in Electrical Engineering in 2020. From 2020 to 2022, he was a postdoctoral researcher in Department of Mechanical and Aerospace Engineering at University of California San Diego. He is now an Assistant Professor in Beijing Institute of Technology, China. His research interest includes switched/hybrid systems and control, stability of nonlinear systems, Lyapunov methods, input-to-state stability theory and matrix perturbation theory.

Sonia Martínez (M'02-SM'07-F'18) is a Professor of Mechanical and Aerospace Engineering at the University of California, San Diego, CA, USA. She received the Ph.D. degree in Engineering Mathematics from the Universidad Carlos III de Madrid, Spain, in May 2002. She was a Visiting Assistant Professor of Applied Mathematics at the Technical University of Catalonia, Spain (2002-2003) and a Postdoctoral Fulbright Fellowship at the Coordinated Science Laboratory of the University of Illinois, Urbana-Champaign (2003-2004) and the Center for Control, Dynamical systems and Computation of the University of California, Santa Barbara (2004-2005). Her research interests include the control of network systems, multi-agent systems, nonlinear control theory, and robotics. She received the Best Student Paper award at the 2002 IEEE Conference on Decision and Control for her work on the control of underactuated mechanical systems and was the recipient of a NSF CAREER Award in 2007. For the paper "Motion coordination with Distributed Information," co-authored with Jorge Cortés and Francesco Bullo, she received the 2008 Control Systems Magazine Outstanding Paper Award. She is the Editor in Chief of the recently launched Open Journal of Control Systems.

Jorge Cortés (M'02, SM'06, F'14) received the Licenciatura degree in mathematics from Universidad de Zaragoza, Zaragoza, Spain, in 1997, and the Ph.D. degree in engineering mathematics from Universidad Carlos III de Madrid, Madrid, Spain, in 2001. He held postdoctoral positions with the University of Twente, Twente, The Netherlands, and the University of Illinois at Urbana-Champaign, Urbana, IL, USA. He was an Assistant Professor with the Department of Applied Mathematics and Statistics, University of California, Santa Cruz, CA, USA, from 2004 to 2007. He is currently a Professor in the Department of Mechanical and Aerospace Engineering, University of California, San Diego, CA, USA. He is the author of Geometric, Control and Numerical Aspects of Nonholonomic Systems (Springer-Verlag, 2002) and co-author (together with F. Bullo and S. Martínez) of Distributed Control of Robotic Networks (Princeton University Press, 2009). He is a Fellow of IEEE and SIAM. His current research interests include distributed control and optimization, network science, nonsmooth analysis, reasoning and decision making under uncertainty, network neuroscience, and multi-agent coordination in robotic, power, and transportation networks.