Stealthy Sensor Attack Detection and Real-Time Performance Recovery for Resilient CPS

Sangjun Kim, Yongsoon Eun, *Senior Member, IEEE*, and Kyung-Joon Park, *Senior Member, IEEE*

***Abstract*—Cyber-physical attacks exploit intrinsic natures of physical systems and can severely damage cyber-physical systems (CPSs) without being detected by the conventional anomaly detector. In this article, based on software-defined networking, we propose a holistic resilient CPS framework that can detect, isolate, and recover from cyber-physical attacks in real time. To show the effectiveness of the proposed framework, we focus on the pole-dynamics attack (PDA), a newly reported stealthy sensor attack that can make the physical system unstable. We develop an efficient detection algorithm for PDA and embed it into the proposed framework. By implementing a testbed, we validate that the proposed framework guarantees resilience of CPS against the PDA.**

***Index Terms*—Cyber-physical systems (CPSs), real-time attack detection, resilience, software-defined networking (SDN).**

I. Introduction

CYBER-PHYSICAL systems (CPSs) are tightly integrated networked control systems, in which physical systems in the real world and control software in the cyber space are connected through networks [1]–[4]. While the network connectivity of CPS significantly enhances efficiency of complex industrial systems [5], this connectivity is vulnerable to malicious attacks that can make the physical systems unstable [6]–[8]. For instance, in 2015, a malicious attacker invaded the corporate network of the Ukraine power grid and executed malware to cause a massive power outage [9].

A cyber-physical attack is defined as a corruption of the control-related data in CPS, which causes faults in the controls of the physical systems. The goal of the attacker is to make the physical systems reach an unstable state, which may result in significant economic loss, physical equipment damage, and human casualties. To guarantee the safety of physical systems under cyber-physical attacks, a physical-knowledge-based anomaly detector is adopted in the computing systems, which is able to detect certain attacks. Still, several stealthy cyber-physical attacks have been recently reported. These include the replay attack [10], zero-dynamics attack (ZDA) [11], pole-dynamics attack (PDA) [12], and covert attack [13].

The covert attack is the most sophisticated one because it requires the physical system model and needs to manipulate both control input signals and sensor measurements at the same time. Hence, it is difficult to realize a covert attack in practice. Meanwhile, the implementation difficulty for the attacker is substantially reduced with the ZDA and PDA because they only need to corrupt the control input signal or sensor measurement, respectively. In particular, the PDA is the most recently reported stealthy sensor attack, for which detection and mitigation methods have not been fully addressed.

In this article, we propose a cyber-physical security framework against the PDA, which can detect, isolate, and recover from the attack in real time. We approach the PDA detection in the network domain; intelligent devices in the network directly inspect the sensor measurement with a control-theoretical PDA detection algorithm. After PDA detection, we implement the following two consecutive network recovery processes; attacker isolation from the network and feedback loop reconstruction between physical and computing systems. To this end, we adopt software-defined networking (SDN) technology. The main contributions of this article are summarized as follows.

1) We propose a *real-time* resilient CPS framework against the PDA. The proposed architecture is a holistic network-wide approach to detect, isolate, and recover from the PDA in real time.
2) We validate the proposed resilient architecture by substantial experiments. In particular, we implement an SDN testbed and carry out empirical study for the real-world performance.

To the best of authors' knowledge, this work is the first study on PDA detection and mitigation that operates in real time. The rest of this article is organized as follows. In Section II, we present related studies on resilient CPS and stealthy cyber-physical attacks. Section III provides the basics of the PDA. In Section IV, we propose a cyber-physical security framework. In Section V, we evaluate the resilience of our framework against the PDA. Section VI concludes this article.

Manuscript received September 17, 2020; revised December 14, 2020; accepted January 4, 2021. Date of publication January 15, 2021; date of current version July 26, 2021. This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2019R1A2C1088092) and in part by the Institute for Information and Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (2014-0-00 065, Resilient Cyber-Physical Systems Research). Paper no. TII-20-4355. *(Corresponding author: Kyung-Joon Park.)*

The authors are with the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology, Daegu 42988, South Korea (e-mail: sjkim@dgist.ac.kr; yeun@dgist.ac.kr; kjp@dgist.ac.kr).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2021.3052182.

Digital Object Identifier 10.1109/TII.2021.3052182

II. Related Work

A. Resilient CPSs

In the control-theoretical domain, a resilient CPS framework is proposed in [14], which has a physical knowledge based detector and a data-driven detector. When a cyber-physical attack is detected on a supervisory controller, this framework replaces the control policy to maintain the stability of the physical system. However, attack isolation on the network is beyond the scope of this research.

In the computing domain, an attack-tolerant CPS design is presented in [15], where there are two strategies to ensure CPS safety, i.e., software restart and trust execution environment on hardware platforms. These two strategies eliminate abnormal tasks by an adversary and provide trusted computation on the computing systems, where each strategy is selected by the dynamics of the physical systems and the specifications of the computing systems. However, the elimination of the abnormal tasks does not indicate attacker isolation on the CPS.

In the network domain, an SDN-based cyber-physical attack countermeasure is proposed in [7], which only provides resilience for a communication-based train control (CBTC) system against fault-data injection. Distributed SDN devices on the CBTC system observe the intrusion attempts of the attacker. Then, the proposed countermeasure swaps the centralized supervisor system to command the emergency braking to the train. The proposed countermeasure only handles false-data injection based on the address resolution protocol spoofing presented in [16].

B. Stealthy Attacks on CPS

The ZDA and PDA are typical system-theoretic stealthy cyber-physical attacks that corrupt control-related information in communication channels [11], [12]. These unidirectional attacks are relatively simple to implement in comparison to other attacks that invade both channels of the feedback control loop while maintaining their stealthiness.

The ZDA targets the zero-dynamics of linear physical systems in the unstable region, and it corrupts the control input signal by adding a ZDA signal to the network. When the ZDA is injected, the states of the physical system are made unstable by the ZDA; however, the sensor measurement from the physical system remains stable because of the stealthiness of the ZDA.

A ZDA detection scheme using a generalized hold method is proposed in [17], which replaces zero-order hold (ZOH) at the actuator. In [18], the authors propose a two-way coding strategy to reveal the stealthiness of the ZDA, which distorts the attack perspectives of feedback control systems. In addition, the ZDA is inherently detected by the actuating saturation of the physical systems; therefore, it is hard to launch the ZDA in a real CPS environment.

The PDA targets the pole-dynamics of linear physical systems in the unstable region and corrupts the sensor measurements by adding the PDA signal to the network. When the PDA is launched, the states of the physical system diverge to infinity,

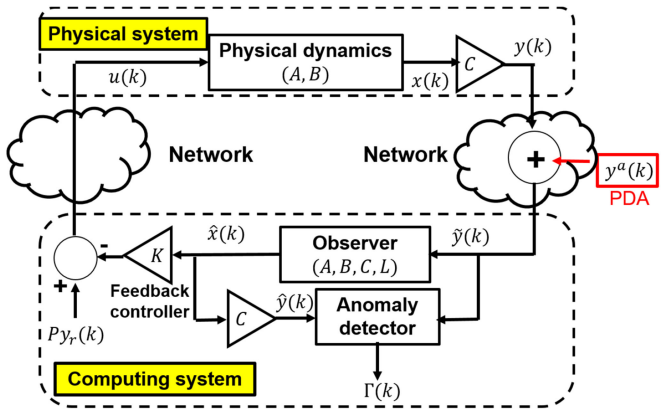


Fig. 1. CPS with a conventional anomaly detector under the PDA.

but the computing system misleads the corrupted sensor measurement into a stable state because the PDA signal vanishes the divergence of the sensor measurement from the physical system.

Unlike the ZDA, there has been little work to detect and recover from the PDA, especially in real time. Consequently, to guarantee resilience of CPS from the PDA, it is required to develop detection and recovery mechanisms.

III. Preliminaries on the PDA

A. System Model

We consider a CPS including a physical system, a computing system, and networks, as illustrated in Fig. 1. The physical system is modeled as a linear time-invariant (LTI) single-input single-output system. The computing system consists of an observer, a feedback controller, and an anomaly detector, where each component estimates the state of the physical system, calculates the control input signal, and determines system anomalies, respectively. The physical system and computing system are designed in the discrete-time domain. The LTI system is one of the most typical models adopted for CPS with physical dynamics such as unmanned aerial vehicles (UAVs) [19] and industrial systems [20].

We consider the dynamics of the physical system given by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$ is the system matrix that has one or more eigenvalues of magnitude greater than 1, $B \in \mathbb{R}^n$ is the input matrix, $C \in \mathbb{R}^{1\times n}$ is the output matrix, $x(k) \in \mathbb{R}^n$ is the state of the physical system, $u(k) \in \mathbb{R}$ is the control input, and $y(k) \in \mathbb{R}$ is the sensor output. For the LTI system (1), we assume that (A, B) pair is controllable and (A, C) pair is observable. The eigenvalues of A whose magnitude is greater than 1 become the unstable poles of the transfer function form of (1) from u to y.

We consider the observer-based feedback controller on the computing system as follows:

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + Bu(k) + L(y(k) - C\hat{x}(k)) \\ u(k) &= Py_r(k) - K\hat{x}(k) \end{aligned} \tag{2}$$

where $\hat{x}(k) \in \mathbb{R}^n$ is the state estimate, $y_r(k) \in \mathbb{R}$ is the reference signal, P is a scalar gain, $K \in \mathbb{R}^{1\times n}$ is the state-feedback controller gain, and $L \in \mathbb{R}^n$ is the observer gain. We assume that K and L in (2) are properly selected to stabilize system (1), i.e., they are selected such that both $A - BK$ and $A - LC$ are Schur matrices. Due to amplitude ratio of the physical system between the reference signal $y_r(k)$ and the steady-state response $y(k)$, the sensor measurement $y(k)$ does not trace the reference signal $y_r(k)$. Therefore, it is required to scale the reference signal $y_r(k)$ with the gain P considering the amplitude ratio of the physical system $C(I_n - A + BK)^{-1}B$. The gain P is selected to satisfy $C(I_n - A + BK)^{-1}BP = 1$ in order to achieve asymptotic tracking for constant y_r, where I_n is the n dimensional unity matrix.

B. State Divergence by the PDA

The goal of the PDA is to diverge the states of the physical system to infinity while sustaining stealthiness to avoid detection. To achieve the goal, the attacker adds a PDA signal to sensor measurement on the network as shown in Fig. 1, which conceals the divergence of sensor measurement from the anomaly detector.

The PDA is an attack on the sensor and is designed using the knowledge of the matrices A and C of the physical systems (1) as follows:

$$\begin{aligned} x^a(k) &= A^k x^a(0) \\ y^a(k) &= -Cx^a(k) \\ \tilde{y}(k) &= y(k) + y^a(k) \end{aligned} \tag{3}$$

where $x^a(k) \in \mathbb{R}^n$ is the state of the attacker, $x^a(0)$ is the initial state of attack dynamics, $y^a(k)$ is the attack signal, and $\tilde{y}(k)$ is the sensor measurement manipulated by the attacker. Consequently, $\tilde{y}(k)$ enters observer (2). Due to the unstable eigenvalues of system (1), the attacker state $x^a(k)$ diverges as k increases, and due to the observability of (A, C), the PDA signal $y^a(k)$ also diverges.

Under the PDA attack, the closed-loop dynamics of (1)–(3) in terms of two newly defined vectors

$$\tilde{x}(k) = x(k) - x^a(k) \tag{4}$$

$$e(k) = \tilde{x}(k) - \hat{x}(k) \tag{5}$$

are given by

$$\tilde{x}(k+1) = (A - BK)\tilde{x}(k) + BKe(k) + BPy_r(k) \tag{6}$$

$$e(k+1) = (A - LC)e(k). \tag{7}$$

Notice first from (7) that

$$\lim_{k\to\infty} e(k) = \lim_{k\to\infty} [\tilde{x}(k) - \hat{x}(k)] = 0 \tag{8}$$

holds for the vector $e(k)$ because $A - LC$ is Schur. This means that the state estimate $\hat{x}(k)$ asymptotically converges to $\tilde{x}(k)$ (instead of $x(k)$) under the PDA. Now, (6) implies that $\tilde{x}(k)$ is bounded because $A - BK$ is Schur, e is diminishing, and $BPy_r(k)$ is bounded. However, $x^a(k)$ is a diverging signal. Thus, in order for $\tilde{x}(k)$ remain bounded, the state of the physical system $x(k)$ has to diverge as well, closely following the attacker's state $x^a(k)$. If there exists an index $l > 0$ such that $y_r(k) = 0$ for all $k \geq l$, then the following also holds:

$$\lim_{k\to\infty} \tilde{x}(k) = \lim_{k\to\infty} [x(k) - x^a(k)] = 0. \tag{9}$$

This again shows that the state $x(k)$ has to diverge under the PDA attack.

C. Stealthiness of the PDA

The conventional anomaly detector determines the control fault of the physical system as shown in Fig. 1 by comparing the sensor measurement and state estimate from the knowledge on the physical system. To demonstrate the stealthiness of the PDA, we consider the physics-based anomaly detector as follows:

$$\Gamma(k) = \begin{cases} 1, & \text{if } r(k) > \delta \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where $\Gamma(k)$ is the attack alarm, $r(k) = \|y(k) - \hat{y}(k)\| \in \mathbb{R}$ is the residual, and $\delta \in \mathbb{R}$ is the threshold. When the residual $r(k)$ exceeds the threshold δ, the anomaly detector (10) raises an alarm.

When deceived sensor measurement $\tilde{y}(k)$ is injected into the anomaly detector, the PDA detection criterion replaces the residual $r(k)$ with residual $\tilde{r}(k) = \|\tilde{y}(k) - \hat{y}(k)\|$, where $\tilde{r}(k)$ is the residual under the PDA.

From the convergence (8), the residual $\tilde{r}(k)$ converges to zero

$$\lim_{k\to\infty} \tilde{r}(k) = \lim_{k\to\infty} \|C\tilde{x}(k) - C\hat{x}(k)\| = 0. \tag{11}$$

Although the state of the physical system $x(k)$ diverges to infinity by (9), the residual $\tilde{r}(k)$ does not exceed the threshold δ. In consequence, the attack alarm $\Gamma(k)$ is not raised, which indicates that the PDA guarantees stealthiness from (11).

D. Vulnerability Analysis in Cyber Domain

In various CPSs, general Internet protocol is adopted to control the physical systems such as UAVs [21] and CBTC systems [22]. Especially, user datagram protocol (UDP) is widely selected to satisfy real-time requirements of the CPSs, which means that CPSs are vulnerable to most network attacks. One example of the network attacks targeting CPS is a man-in-the-middle (MITM) attack on the CBTC system [7], which shows that the manipulation of the control input packet leads to train collision. While encryption techniques prevent the MITM attacks in conventional networks, it is often difficult to adopt the encryption techniques in CPS due to limited computing resources and real-time requirements of the physical systems.

Cyber-physical attacks, including the PDA, are implemented as packet manipulation by the MITM attacks or intrusion of an unauthorized network device with packet handling software such as NetFilterQueue [23] as illustrated in Fig. 2(a). In this article, the PDA is realized as packet manipulation by the unauthorized device illustrated in Fig. 2(a), and we assume that the packet manipulation is due to a network attack that exploits certain network vulnerabilities. Thus, it is impossible to detect the PDA with a legacy network intrusion detection system.

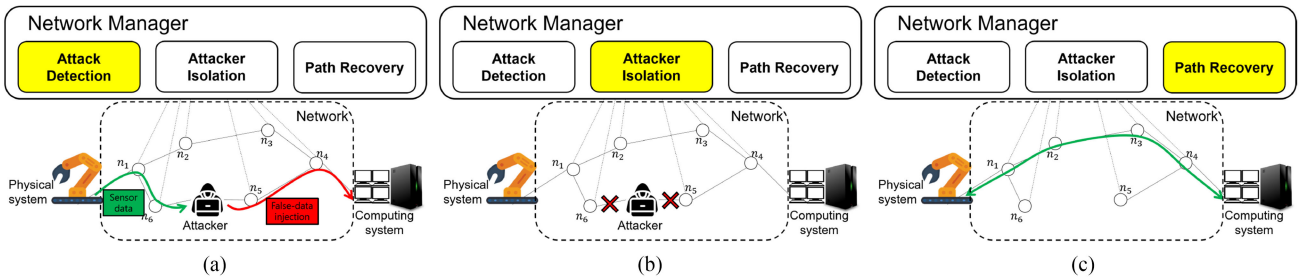


Fig. 2. Three processes of the proposed resilient framework against PDA. (a) Attack detection. (b) Attacker isolation. (c) Path recovery.

IV. SDN-Based Resilient CPS Architecture

In this section, we propose a cyber-physical security framework based on SDN.

A. Real-Time PDA Detection and Recovery Processes

Due to the stealthiness of the PDA, it is impossible to detect the PDA by the conventional anomaly detector at the computing system. In the meantime, in order to realize the PDA, the attacker needs to invade a certain point in the network to corrupt the sensor measurement. Consequently, we tackle the problem from a networking point of view, which enables us to derive a solution for detecting, isolating, and recovering from the attack.

We propose a cyber-physical security framework that supports PDA detection and system recovery in real time as shown in Fig. 2, which consists of a centralized network manager and multiple distributed network devices. The proposed framework executes the following three processes.

1) Attack detection: As shown in Fig. 2(a), distributed network devices inspect the sensor measurement packets and detect the PDA by the PDA detection algorithm embedded on the network devices. Then, the devices detecting the PDA send alarms to the network manager.
2) Attacker isolation: Once the network manager receives attack notification from certain network devices, it identifies the point of attack by comparing attack alarms and a previously acquired network topology, as shown in Fig. 2(b). Once the point of attack is located, the network manager disconnects the links next to the attacker to prevent additional packet corruptions by the attacker.
3) Path recovery: Due to attacker isolation, the original connection between the physical system and the computing system is lost, which indicates destruction of the feedback control loop. To recover the feedback loop, the network manager searches and establishes a new path between the physical system and computing system as shown in Fig. 2(c).

B. SDN-Based Implementation of the Proposed Architecture

When the PDA is launched, the proposed framework executes these three processes sequentially. A critical issue is to complete these processes before the physical systems enter an irreparable state as a result of the attack. Since legacy network devices have limited programmable resources, it is formidable to implement these processes in a conventional network. Therefore, we adopt the SDN technology to implement the proposed framework.

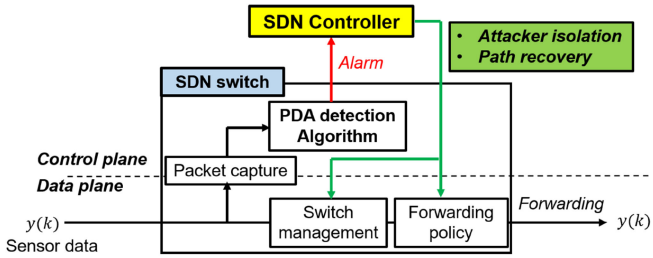


Fig. 3. Framework implementation on the SDN switch.

SDN separates control plane and data plane in software, which is composed of a centralized SDN controller on the control plane and multiple SDN switches on the data plane [24]. The separated control plane and data plane interact through a southbound API, such as OpenFlow, which enables flexible routing and network status monitoring. Both the control plane and data plane of the SDN are programmable by the network administrator. Hence, the SDN switch can utilize the computational resources for cyber-physical security tasks.

Fig. 3 shows the implementation and operation of the proposed framework on the SDN network, where three core processes of the proposed framework are implemented by the SDN network providing services that the legacy network cannot provide as follows.

1) Packet capturing: SDN switch enables us to collect both legitimate sensor packets $y(k)$ and modified sensor packets $\tilde{y}(k)$ by the attacker; however, the anomaly detector (10) on the computing system only acquires the modified sensor measurement $\tilde{y}(k)$.
2) PDA detection logic installation: Programmability of the SDN switch enables us to embed the PDA detection algorithm on the SDN switch, which reveals the PDA from the collected sensor measurement packets.
3) Interaction between the SDN controller and the SDN switches: The SDN switches can detect the PDA and alert to the SDN controller in real time, as illustrated in Fig. 3. Also, the SDN controller remotely manages the SDN switches and provides flexible routing of the network, which supports the attacker isolation process and path recovery process.

In addition, since real-time inspection of the control input signal as well as sensor measurement is possible, the SDN switch can detect other stealthy attacks by embedding different attack detection algorithms.

The SDN switches on the data plane are programmable [24]; the SDN switch is one of the Linux machine, and the packet forwarding function is implemented by software. Hence, the SDN switch can utilize the computational resources for cyber-physical security tasks, excluding networking resources. The attack detection process of the proposed framework is fulfilled by the SDN switches. By utilizing the computational resources, the SDN switches are able to capture the packets for traffic monitoring [25]. Then, based on the captured sensor measurement packets, the SDN switches detect the PDA by the detection algorithm. When the PDA is detected, the SDN switches inform the SDN controller of the attack as shown in Fig. 3.

The attacker isolation process is conducted on the SDN control plane. Based on the topology information and the attack alarms received from the SDN switches, the SDN controller can identify the location of the attacker and adjacent SDN switches linked to the attacker. To isolate the attacker from the network, the SDN controller deactivates the ports of the SDN switches that establish the poisoned link by the attacker through remote switch management as illustrated in Fig. 3. By this port deactivation on the control plane, it is no more possible for the attacker to inject malicious packets and manipulate the legitimate packets on the network.

The path recovery process is conducted with the flexible routing of the SDN control plane. Due to the port deactivation to isolate the attacker, it is necessary to re-establish the data transmission path for sensor measurement packets. The SDN controller updates the network topology and restructures a new data transmission path through the installation of new forwarding policies on the SDN switches, as shown in Fig. 3.

In order to apply the proposed framework to an actual industrial system, the existing network devices should be replaced by SDN switches that contain PDA detection logic. Also, the SDN controller needs to be added to the network, where applications for estimation of attacker location and path recovery are installed on the SDN controller.

C. PDA Detection Algorithm on SDN Switch

We embed the PDA detection algorithm on the SDN switches, where the proposed detection algorithm classifies the PDA from other anomalies such as disturbances and transient behaviors. Here, we assume the following attack location conditions; the PDA is only launched on the network and the link between the physical systems and the SDN switches are protected from the attacker. Then, based on the truthful sensor measurement tracing the open-loop response under the PDA, we develop a dual anomaly detector as the PDA detection algorithm embedded on the SDN switches.

Fig. 4 illustrates the structure of the proposed dual anomaly detector, which consists of a closed-loop model-based observer and an open-loop model observer.

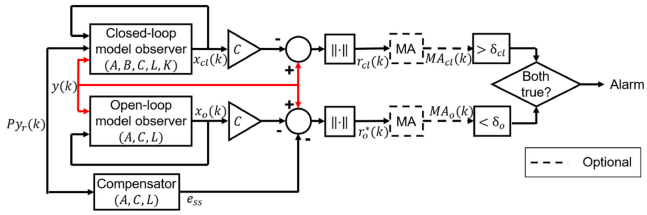


Fig. 4. Dual anomaly detector for PDA detection.

1) Dual Anomaly Detector With Two Observers: We consider the closed-loop model-based observer and open-loop model observer as follows:

$$\hat{x}_{\mathrm{cl}}(k+1) = (A - BK)\hat{x}_{\mathrm{cl}}(k) + BPy_r(k) + L(\tilde{y}(k) - C\hat{x}_{\mathrm{cl}}(k)) \tag{12}$$

$$\hat{x}_o(k+1) = A\hat{x}_o(k) + L(\tilde{y}(k) - C\hat{x}_o(k)) \tag{13}$$

where $\hat{x}_{\mathrm{cl}}(k) \in \mathbb{R}^n$ is the state estimate of the closed-loop model observer (12), and $\hat{x}_o(k) \in \mathbb{R}^n$ is the state estimate of the open-loop model observer (13).

The proposed dual anomaly detection adopts the anomaly detector strategy (10) with two residuals $r_{\mathrm{cl}}(k) = \|\tilde{y}(k) - C\hat{x}_{\mathrm{cl}}(k)\|$ and $r_o(k) = \|\tilde{y}(k) - C\hat{x}_o(k)\|$. In an attack-free system, i.e., when there is no attack, the residual $r_{\mathrm{cl}}(k)$ converges to zero because observer (12) on the SDN switches is equal to observer (2) on the computing system. However, similarly to the gain P selection in (2), the estimated sensor measurement $C\hat{x}_o(k)$ does not trace the constant reference y_r due to the amplitude ratio of the open-loop observer $C(I_n - A + LC)^{-1}L$. Therefore, residual $r_o(k)$ converges to the steady-state deviation e_{ss} between the sensor measurement $\tilde{y}(k)$ and the sensor measurement estimate $C\hat{x}_o(k)$, given by

$$e_{ss} = (1 - C(I_n - A + LC)^{-1}L)y_r \tag{14}$$

for the constant y_r.

Note that, because the deviation e_{ss} is linearly proportional to the reference signal $y_r(k)$, it is not straightforward to adopt the residual $r_o(k)$ for the threshold-based detection strategy (10). Therefore, compensating the deviation e_{ss} in the residual $r_o(k)$ is required, as shown in Fig. 4.

2) Compensation for Model Inaccuracy: The idea of compensation relies on the value of e_{ss} in (14), which is obtained as follows. When there is no attack, and for constant y_r, the open-loop observer estimate $\hat{x}_o(k)$ converge to a constant vector

$$\hat{x}_o(k+1) = \hat{x}_o(k) = \hat{x}_o^* \tag{15}$$

where $\hat{x}_o^* \in \mathbb{R}^n$ is the state estimate of (13) in the steady state. By (6) and the selection of P, the sensor measurement $\tilde{y}(k)$ asymptotically tracks the reference signal $y_r(k)$, and the reference signal $y_r(k)$ has a constant value y_r. From the steady-state condition (15), the state estimation $\hat{x}_o^*$ of observer (13) is represented as

$$\hat{x}_o^* = (I_n - A + LC)^{-1}Ly_r. \tag{16}$$

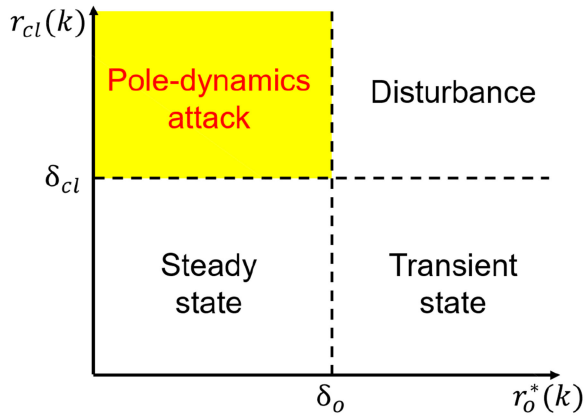


Fig. 5. PDA detection algorithm with dual observers.

Algorithm 1: Identification of the Attacker Link.

Require: P
  $e_{attack} \leftarrow \emptyset, T \leftarrow \emptyset$
  Receive the attack alarms from V
  for $i = 1$ to $|V|$ **do**
    if $v_i.alarm = True$ **then**
      T.append(v_i)
    end if
  end for
  for $j = 1$ to $|E|$ **do**
    if $e_j.v_{source} \in T$ and $e_j.v_{sink} \notin T$ **then**
      $e_{attack} \leftarrow e_j$
    end if
  end for
  return e_{attack}

The value of e_{ss} is in fact the tracking error between y_r and the sensor output estimate $C\hat{x}_o^*$ of observer (13), which gives the equation of (14).

From the steady-state deviation e_{ss}, we replace the residual $r_o(k)$ with compensated residual $r_o^*(k) \in \mathbb{R}$ denoted as

$$r_o^*(k) = \|\tilde{y}(k) - C\hat{x}_o(k) - e_{ss}\|. \tag{17}$$

The residual $r_o^*(k)$ converges to zero in the attack-free steady state, which is the criterion to filter the PDA from the other anomalies such as disturbances.

3) Stealthiness Revealed by Dual Anomaly Detector: Fig. 5 presents the state determination of the physical system by the dual anomaly detector with two residuals $r_{\text{cl}}(k)$ and $r_o^*(k)$, from which we can filter out the PDA from external disturbances.

In an attack-free system, the residuals $r_{\text{cl}}(k)$ and $r_o^*(k)$ converge to zero. When a disturbance is applied to the physical system, the residuals $r_{\text{cl}}(k)$ and $r_o^*(k)$ exceed the thresholds δ_{cl} and δ_o, respectively. The transient state refers to the period in which the physical system recovers from the anomalies to the attack-free state. For an example of the transient state, after the disturbance injection, the residuals $r_{\text{cl}}(k)$ and $r_o^*(k)$ recover to zero, where the $r_{\text{cl}}(k)$ converges faster than $r_o^*(k)$ because the closed-loop model observer (12) has better model fidelity than the open-loop model observer (13). However, when the PDA is launched on the network, then only the residual $r_{\text{cl}}(k)$ diverges to infinity, although the residual $r_o^*(k)$ converges to zero by (3) and (9). Consequently, the proposed dual anomaly detector can filter out and detect the PDA from other abnormalities.

To prevent the false-positive detection by system noises or external disturbances, the moving average (MA) can be adopted optionally for the residuals as shown in Fig. 4. Tradeoff between false-positive detection rate and detection speed by MA is described in Section V.

D. Attacker Isolation and Path Recovery

On the SDN controller, a delivery path of the sensor measurement is denoted as the directed graph $P = \{V, E\}$ representing the set of the SDN switches and links, where V is a set of the SDN switches, and E is a set of the links. For the set of links $E = \{e_1, e_2, \ldots, e_{|V|-1}\}$, the directed ith link $e_i = \{v_{\text{source}}, v_{\text{sink}}\}$ on the path P is denoted by the set of two SDN switches, where v_{source} is the source switch of the link, and v_{sink} is the sink switch of the link. In the link e_i, the data are transmitted from v_{source} to v_{sink}.

If the attacker invades any link on the path P and launches the PDA, still some SDN switches receive the truthful sensor measurement from the physical system and the others receive the manipulated sensor measurement by the attacker. Then, the SDN controller receives attack alarms from SDN switches and locates the attacked link by Algorithm 1.

In Algorithm 1, T is a set of SDN switches that receive the truthful sensor measurement, T.append(v) is a function that includes an SDN switch v in set T, v_i.alarm is an indicator whether the SDN switch v_i generates the attack alarm, $e_j.v_{\text{source}}$ and $e_j.v_{\text{sink}}$ are the data transmission source SDN switch and the sink SDN switch in the directed link e_j, respectively. Algorithm 1 classifies the SDN switches that receive the truthful sensor measurement into the set T from all the switches V on the path P. Then, the estimation algorithm analyzes the SDN switches T for all links E. For example, if the attacker invades the link $e_{\text{attack}} = \{a, b\}$, then an SDN switch a receives the truthful sensor measurement $y(k)$ and alerts the SDN controller about the PDA. However, the other SDN switch b receives the corrupted sensor measurement $\tilde{y}(k)$ from the attacker and does not send an alarm.

To minimize additional damage by the PDA, the attacker should be isolated from the network. The SDN controller sends a command to the SDN switches between the attacker's link e_{attack}, and then these SDN switches deactivate the ports that establish the link e_{attack}. Then, the link e_{attack} is disconnected, and no more data are transmitted to the attacker.

Due to the attacker isolation, a new data transmission path for sensor measurement is required. After the recovery of the transmission path, the computing system receives the truthful sensor measurement and recovers the physical system to the original state.

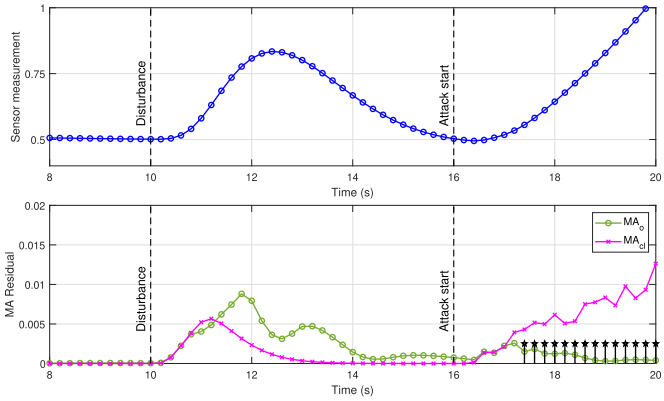


Fig. 6. PDA detection by the dual anomaly detector.

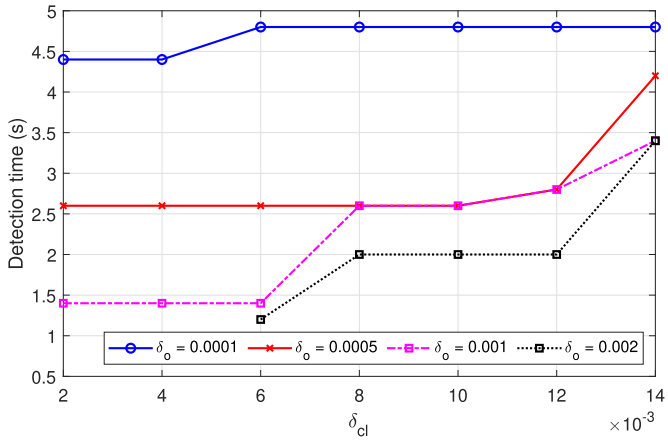


Fig. 7. PDA detection time with respect to threshold assignments.

V. Performance Evaluation

A. Physical System Model for Recovery Demonstration

We consider the well-known ball and beam control system as the physical system [26], which consists of a beam with an adjustable tilt and a ball to roll off from the beam by gravity; therefore, this system has inherently unstable poles. The goal of the control of this system is to regulate the position of the ball on the beam by tilting the beam. We set the range of the beam between 0 and 1 m and the reference position is $y_r(t) = 0.5$ m. If the position of the ball exceeded the range of the beam, we consider the physical system irreparable because the ball rolls off the beam.

We consider the linearized ball-beam system dynamics in continuous time domain as follows:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-mg}{\frac{J}{R^2}+m} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u(t)$$
$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}(t) \tag{18}$$

where $\mathbf{x}(t) = [d(t)\ \dot{d}(t)\ \theta(t)\ \dot{\theta}(t)]^T$ is a state vector of the ball-beam system, $d(t)$ is the ball position, $\theta(t)$ is the beam angle, m is the mass of the ball, g is gravitational acceleration, J is the ball's moment of inertia, and R is the radius of the ball. In (18), the pair of the system matrix and the input matrix satisfies the controllability, and the pair of the system matrix and the output matrix satisfies the observability [27]. The system form (1) can be obtained by discretization of system (18) with ZOH.

B. Simulation Results for PDA Detection Performance

1) Detectability of the PDA: Fig. 6 shows the simulation result of the PDA detection algorithm on SDN switches that receive the truthful sensor measurement, where we apply a short disturbance to the physical system at $t = 10$ s and launch the PDA to the computing system at $t = 16$ s to validate the classification between the disturbance and the PDA.

Although more detection time is required, we adopt MA on our proposed detection algorithm for trustworthy PDA detection against the external disturbance or system noise. The top graph of Fig. 6 shows the sensor measurement $y(k)$, and the bottom graph shows the filtered residuals $MA_{\text{cl}}(k)$ and $MA_o(k)$, where the residuals $r_{\text{cl}}(k)$ and $r_o^*(k)$ are replaced by MAs with the time-window size $w = 5$. In addition, the black stemplot in the bottom graph shows the PDA detection by the residuals $MA_{\text{cl}}(k)$ and $MA_o(k)$ at time step k.

Due to the disturbance at $t = 10$ s, two residuals increase simultaneously as the sensor measurement $y(k)$ increases. When the PDA is launched, only the residual $MA_o(k)$ converges to zero while the residual $MA_{\text{cl}}(k)$ diverges to infinity. At $t = 17.4$ s, the SDN switch detects the PDA, where the graph in the bottom part of Fig. 6 indicates attack detection. The simulation results clearly show that our proposed PDA detection algorithm can detect the PDA before the physical system becomes unstable by successfully discriminating the PDA from disturbances.

2) Detection Performance by Threshold Assignments: Our proposed PDA detection algorithm requires two thresholds δ_{cl} and δ_o, hence, the assignments of these thresholds are crucial for the attack detection performance. The assignments of these thresholds are dependent on the MA window size and classification of four states of the dual anomaly detector.

In the context of the disturbance decision, the threshold δ_{cl} should be small enough to detect the disturbances, i.e., the disturbance cannot be detected when too large threshold δ_{cl} is selected. As shown in Fig. 6, the threshold δ_{cl} does not exceed 6×10^{-3} under the disturbance. Therefore, the threshold δ_{cl} must be selected less than 6×10^{-3} to detect the disturbance. In addition, the threshold δ_o is selected to classify the state of the physical system into the transition state and the steady state. Under the recovery process from the disturbance, as illustrated in Fig 6, the sensor measurement $y(t)$ stays within 25% range of the reference y_r when $\delta_o = 1 \times 10^{-3}$. However, in case of the threshold $\delta_o = 1 \times 10^{-4}$, the sensor measurement $y(t)$ stays within 3% range of the reference y_r.

In the viewpoint of the PDA detection time and false-positive detection, the thresholds δ_{cl} and δ_o are adjusted. Fig. 7 illustrates

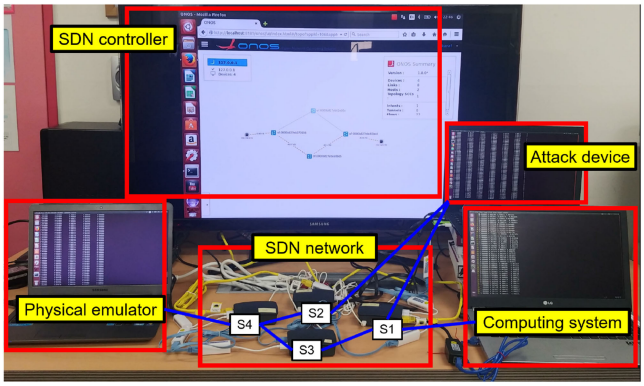


Fig. 8. Testbed configuration.

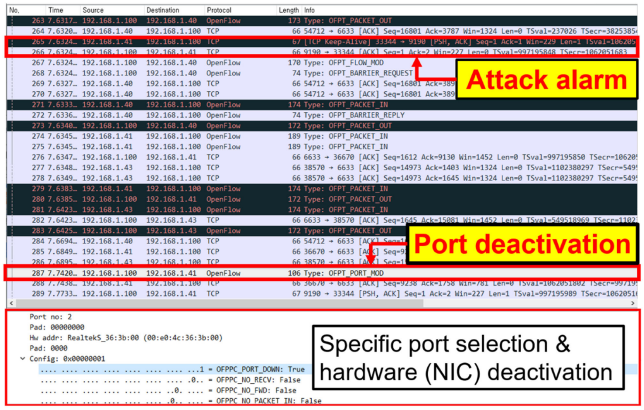


Fig. 9. Port deactivation of the SDN controller under PDA.

the simulation results for the PDA detection time with respect to the threshold assignments. The increase in the threshold δ_o reduces the PDA detection time as shown in Fig. 7. However, on the other hand, which provides a poor performance for classification between the steady state and the transient state, as illustrated in Fig. 6. The decrease in the threshold δ_{cl} also reduces the PDA detection time as shown in Fig. 7, while it causes the false-positive detection of the proposed detection algorithm for disturbances. When the threshold $\delta_o = 2 \times 10^{-3}$, the proposed PDA detection algorithm shows false-positive detection with the threshold $\delta_{\text{cl}} < 6 \times 10^{-3}$ from the threshold assignments simulation result, as illustrated in Fig. 7.

Consequently, both thresholds δ_{cl} and δ_o should be assigned by considering disturbance determination, attack detection speed, and false-positive rate. In our testbed environment, the threshold δ_{cl} should be assigned less than 6×10^{-3} from the viewpoint of disturbance detection. At the same time, the threshold δ_o should be selected less than 1×10^{-3} in the context of the PDA detection time and sensitivity.

C. Testbed Environment

We implement a CPS testbed with SDN. The testbed consists of a physical emulator, a computing system, an SDN network, and a PDA attacker, as shown in Fig. 8. The physical emulator is a PC, which emulates ball and beam dynamics governed by the control input signal from the computing system. The computing system is implemented in a PC with the control software including the observer, anomaly detector, and feedback controller. The physical emulator and computing system are connected through the SDN, which constructs a feedback control loop. The physical emulator periodically transmits the sensor measurement to the computing system, which then calculates the control input and sends it to the physical emulator via SDN. To implement the attack device, we employ the NetFilterQueue [23] software on a PC. We connect the attack device between the SDN switches $S1$ and $S2$, as shown in Fig. 8. Therefore, the sensor measurement packets transmitted through the attack device are manipulated by adding the PDA signal (3). We adopt general UDP protocol for communication between the physical emulator and the computing system to satisfy the real-time control requirement of the control system.

The SDN consists of an SDN controller and four SDN switches. The SDN switches are implemented by Raspberry pi 3 s with OpenvSwitch software [28]. We embed the PDA detection algorithm in the SDN switches. We use the ONOS SDN controller [29] to manage the SDN switches and reroute the packets when an attack is detected. When the sensor measurement packet enters the attacker, the attacker corrupts the packet by adding the PDA signal and redirects the modified packet to the computing system.

We consider the following scenario for experiments.

1) The attacker knows the sensor measurement packet flow $S4$–$S2$–$S1$ and invades the link between $S1$ and $S2$. Then, the attacker launches the PDA on the link by corrupting sensor measurement packet.
2) The SDN switches $S2$ and $S4$ inspect the truthful packet, and detect the PDA by the embedded dual anomaly detector. After PDA detection, these SDN switches notify the SDN controller of the attack.
3) Then, the SDN controller identifies the location of the attacker by Algorithm 1 and sends a command to cut-off the link to SDN switches $S1$ and $S2$ in the attacker isolation process.
4) The SDN controller constructs a new sensor measurement packet flow $S4$–$S3$–$S1$ in the path recovery process. Consequently, the computing system receives the truthful sensor measurement $y(k)$ and recovers the physical system from the damage of the PDA.

To enhance the understanding of the SDN process, we provide the details on how the SDN manages the communication packets. In the attacker isolation process, the SDN controller receives the attack alarms from the SDN switches, and the SDN controller disconnects the links of the attacker. Fig. 9 shows packet traces on the SDN controller, where the SDN controller reacts to the PDA detection alarm for attacker isolation. When the attack alarms are entered to the SDN controller, the SDN controller estimates the attack location, and sends OpenFlow command OFPT_PORT_MOD, which modifies the behavior of the port, to SDN switch $S2$, as shown in Fig. 8. Then, the SDN switch $S2$ deactivates the network interface card (NIC) corresponding to the OFPT_PORT_MOD command. Due to the deactivated NIC,

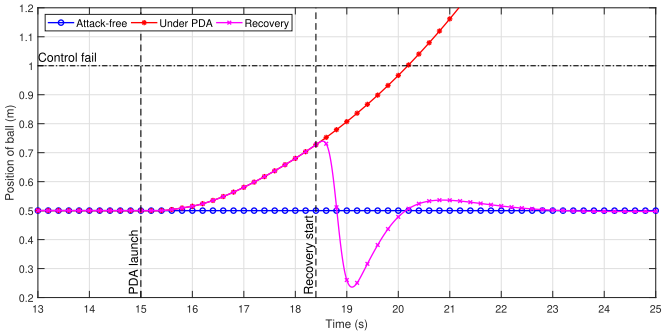
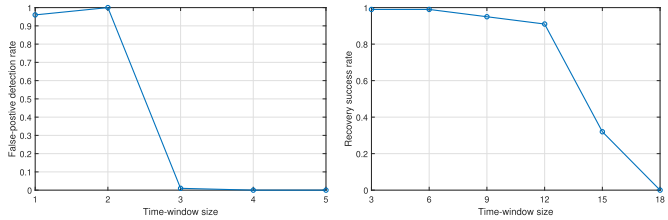


Fig. 10. Sensor measurements of recovery experiment.



(a) (b)

Fig. 11. Evaluation of PDA detection algorithm. (a) False-positive detection rate. (b) Recovery success rate.

it is denied for sensor measurement packets to pass through the attacker and for the attacker to inject malicious packets.

D. Empirical Study on Performance Recovery

We conduct two types of experiments to evaluate the reliability and recovery of our framework, where we select the following three evaluation metrics: false-positive detection rate, recovery success rate, and recovery performance. In the reliability experiment, the false-positive detection rate quantifies the reliability of attack detection and control quality deterioration in an attack-free environment. In the recovery experiment, the recovery success rate and recovery performance quantify the resilience of the CPS under the PDA. By changing the time-window size w in the PDA detection algorithm, we evaluate the three metrics in the experiments with fixed thresholds δ_{cl} and δ_o.

1) Experiment on Reliability: To evaluate the false-positive detection rate, we conduct a reliability experiment in an attack-free environment. We only apply a short disturbance to the physical emulator and check whether attack alarms occur or not at the SDN switches. By changing the time-window size w, we conduct experiment 100 times for each time-window size w to count the false-positive detection alarms at the SDN switches.

2) Experiment on Recovery: We conduct a recovery experiment to evaluate the recovery success rate and recovery performance under the PDA. The physical emulator runs for $t_f = 25$ s, and the attacker launches the PDA to the computing system at attack start time $t_a = 15$ s. We check whether the position of the ball exceeds the range of the beam under the PDA by changing the time-window size w.

Fig. 10 presents the empirical result of the recovery experiments for our framework when the time-window size w is 3, which compares the recovered sensor measurements $y(t)$ with cases of the attack-free and attacked without recovery. In the attack-free case, the position of the ball $y(t)$ is regulated to 0.5 m. Under the PDA, the position of the ball $y(t)$ diverges to infinity from the time $t_a = 15$ s; then the ball exceeds the beam at time $t = 20.2$ s. However, for the case of our proposed framework, the physical system enters the recovery process at $t = 18.4$ s; consequently, the sensor measurement $y(t)$ quickly converged to 0.5 m.

The recovery performance is evaluated in terms of the integrated absolute error (IAE), which is the integral of the absolute value of error between sensor measurement $y(t)$ and the position reference $y_r(t)$ from the attack start time t_a. The IAE is calculated as follows:

$$\text{IAE} = \int_{t_a}^{t_f} |y(t) - y_r(t)|\, dt \tag{19}$$

where a larger IAE indicates the worse recovery performance. Similarly to the reliability experiment, we conduct the recovery experiment 100 times for each time-window size w by changing the time-window size w, and we check the number of recovery success cases and IAEs in all trials.

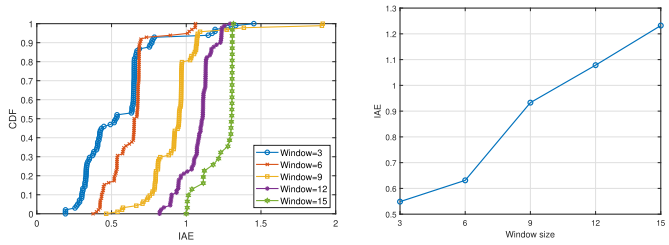
E. Experimental Results

1) False-Positive Detection Rate and Recovery Success Rate: Fig. 11 shows the false-positive detection rate and recovery success rate with respect to time-window size w. The false-positive detection rate and recovery success rate indicate the noise sensitivity and the attack detection speed of PDA detection algorithm, respectively.

Fig. 11(a) shows decrease of the false-positive detection rate as the time-window size w increases. In particular, when the time-window size w is 3, the false-positive detection rate drastically decreases, and the detection algorithms with w greater than 3 do not generate false-positive alarms. In other words, too small w makes the detection algorithm vulnerable to noises, which causes the network instability and poor control performance due to false detection alarms.

Fig. 11(b) shows the decrease in the recovery success rate as the time-window size w increases, where we set the initial time-window size w as 3 to guarantee the detection reliability from the results of the reliability experiments. The dominant reason for recovery fail is detection delay by MA with a large time-window size w at the SDN switches. For the time-window size $w = 15$, the recovery success rate drastically decreases to 0.32; in addition, the physical emulator becomes irreparable in all cases in which the time-window size $w = 18$.

2) Recovery Performance: We analyze the IAEs only for the recovery success cases in the recovery experiment results for each time-window size w. Fig. 12(a) presents the cumulative distribution functions (CDFs) of the IAEs, which show the recovery performances of empirical trials for each time-window size w, where the PDA detection algorithm with a large time-window size w causes a poor recovery performance of the physical system. In most cases, small time-window sizes show a better



(a) (b)

Fig. 12. Integrate absolute error in recovery processes. (a) CDF of recovery performances. (b) Mean recovery performances.

recovery performance than large-window sizes. However, IAEs at $w = 6$ are less than IAEs at $w = 3$ in some cases for a poor recovery performance, because small w is more sensitive to noise than large w, which degrades the attack detection performance. Fig. 12(b) shows the average IAEs of the recovery experiment for each time-window size, which show that a large time-window size generally causes a poor recovery performance under the PDA.

3) Time-Window Size Selection: From the reliability and recovery experiments, we confirm that there exists a tradeoff between recovery performance and detection reliability by the time-window size w. To minimize damage by the PDA, it is necessary to select a suitable time-window size w. The lower bound of the window size range should be large enough to have a low false-positive detection rate for noise-tolerant detection. Likewise, the upper bound should be small enough to ensure a high-recovery success rate for noise-tolerant detection. In our testbed environment, the lower bound of w is greater than 2 for noise-tolerant detection. In addition, the upper bound of w was less than 9 to achieve a recovery success rate over 95%.

VI. Conclusion

In this article, we have proposed a cyber-physical security framework from the networking perspective, which guarantees the resilience of CPS against the PDA, a newly reported stealthy sensor attack. In order to validate the proposed framework, we have implemented a testbed with a physical emulator and an SDN. Our empirical results have shown that the proposed framework can ensure the resilience of a CPS against the PDA in real time.

The computing aspect of SDN switches needs further research. The SDN switches actually have limited computational resources, and the more computational resources are used for packet inspection, the lower the network performance is, which leads to a decrease in the control performance of the physical system. As a line of future work, one possibility is to investigate the energy-efficient packet sampling strategy maintaining the resilience of the physical systems while considering the energy constraints of the SDN switches. Another possibility is to extend this work to an environment where there are multiple physical systems with heterogeneous dynamics.

References

[1] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: The next computing revolution," in *Proc. Des. Autom. Conf.*, 2010, pp. 731–736.
[2] K.-J. Park, J. Kim, H. Lim, and Y. Eun, "Robust path diversity for network quality of service in cyber-physical systems," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2204–2215, Nov. 2014.
[3] K.-J. Park, R. Zheng, and X. Liu, "Cyber-physical systems: Milestones and research challenges," *Comput. Commun.*, vol. 36, no. 1, pp. 1–7, 2012.
[4] D. Kim, Y. Won, S. Kim, Y. Eun, K.-J. Park, and K. H. Johansson, "Sampling rate optimization for IEEE 802.11 wireless control systems," in *Proc. 10th ACM/IEEE Int. Conf. Cyber-Phys. Syst.*, 2019, pp. 87–96.
[5] F. Farivar, M. S. Haghighi, A. Jolfaei, and M. Alazab, "Artificial intelligence for detection, estimation, and compensation of malicious attacks in nonlinear cyber-physical systems and industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2716–2725, Apr. 2020.
[6] A. Teixeira, D. Pérez, H. Sandberg, and K. H. Johansson, "Attack models and scenarios for networked control systems," in *Proc. 1st Int. Conf. High Confidence Netw. Syst.*, 2012, pp. 55–64.
[7] S. Kim, Y. Won, I.-H. Park, Y. Eun, and K.-J. Park, "Cyber-physical vulnerability analysis of communication-based train control," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6353–6362, Aug. 2019.
[8] Y. Liu, A. Liu, X. Liu, and M. Ma, "A trust-based active detection for cyber-physical security in industrial environments," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6593–6603, Dec. 2019.
[9] R. Khan, P. Maynard, K. McLaughlin, D. Laverty, and S. Sezer, "Threat analysis of blackenergy malware for synchrophasor based real-time control and monitoring in smart grid," in *Proc. 4th Int. Symp. ICS SCADA Cyber Secur. Res.*, 2016, pp. 53–63.
[10] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *Proc. 47th Annu. Allerton Conf. Commun., Control, Comput.*, 2009, pp. 911–918.
[11] G. Park, C. Lee, H. Shim, Y. Eun, and K. H. Johansson, "Stealthy adversaries against uncertain cyber-physical systems: Threat of robust zero-dynamics attack," *IEEE Trans. Autom. Control*, vol. 64, no. 12, pp. 4907–4919, Dec. 2019.
[12] H. Jeon and Y. Eun, "A stealthy sensor attack for uncertain cyber-physical systems," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6345–6352, Aug. 2019.
[13] R. S. Smith, "Covert misappropriation of networked control systems: Presenting a feedback structure," *IEEE Control Syst. Mag.*, vol. 35, no. 1, pp. 82–92, Feb. 2015.
[14] K. Paridari, N. O'Mahony, A. E.-D. Mady, R. Chabukswar, M. Boubekeur, and H. Sandberg, "A framework for attack-resilient industrial control systems: Attack detection and controller reconfiguration," *Proc. IEEE*, vol. 106, no. 1, pp. 113–128, Jan. 2017.
[15] F. Abdi, C.-Y. Chen, M. Hasan, S. Liu, S. Mohan, and M. Caccamo, "Preserving physical safety under cyber attacks," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6285–6300, Aug. 2018.
[16] Y. Won *et al.*, "An attack-resilient CPS architecture for hierarchical control: A case study on train control systems," *Computer*, vol. 51, no. 11, pp. 46–55, 2018.
[17] J. Back, J. Kim, C. Lee, G. Park, and H. Shim, "Enhancement of security against zero dynamics attack via generalized hold," in *Proc. IEEE 56th Annu. Conf. Decis. Control*, 2017, pp. 1350–1355.
[18] S. Fang, K. H. Johansson, M. Skoglund, H. Sandberg, and H. Ishii, "Two-way coding in control systems under injection attacks: From attack detection to attack correction," in *Proc. 10th ACM/IEEE Int. Conf. Cyber-Phys. Syst.*, 2019, pp. 141–150.
[19] H. Choi *et al.*, "Detecting attacks against robotic vehicles: A control invariant approach," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2018, pp. 801–816.
[20] J. Liu and J. Lin, "Design optimization of WirelessHART networks in cyber-physical systems," *J. Syst. Architecture*, vol. 97, pp. 168–184, 2019.
[21] A. Koubaa, A. Allouch, M. Alajlan, Y. Javed, A. Belghith, and M. Khalgui, "Micro air vehicle link (MAVLink) in a nutshell: A survey," *IEEE Access*, vol. 7, pp. 87658–87680, 2019.
[22] R. He *et al.*, "High-speed railway communications: From GSM-R to LTE-R," *IEEE Veh. Technol. Mag.*, vol. 11, no. 3, pp. 49–58, Sep. 2016.
[23] Kerkhoff Technol. Inc. NetFilterQueue, 2011. [Online]. Available: https://pypi.org/project/NetfilterQueue/
[24] F. Hu, Q. Hao, and K. Bao, "A survey on software-defined network and OpenFlow: From concept to implementation," *IEEE Commun. Surv. Tut.*, vol. 16, no. 4, pp. 2181–2206, Oct.–Dec. 2014.
[25] S. Yoon, T. Ha, S. Kim, and H. Lim, "Scalable traffic sampling using centrality measure on software-defined networks," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 43–49, Jul. 2017.

[26] J. Li, Y. Xia, X. Qi, and Z. Gao, "On the necessity, scheme, and basis of the linear-nonlinear switching in active disturbance rejection control," *IEEE Trans. Ind. Electron.*, vol. 64, no. 2, pp. 1425–1435, Feb. 2017.
[27] C.-T. Chen, *Linear System Theory and Design*, 3rd ed., New York, NY, USA: Oxford Univ. Press, Inc., 1998.
[28] L. Foundation., OpenvSwitch. Jul. 2009. [Online]. Available: http://openvswitch.org/
[29] O. N. Lab., ONOS. Dec. 2014. [Online]. Available: https://wiki.onosproject.org/

Sangjun Kim received the B.S. degree in electronics engineering from Pukyong National University, Busan, South Korea, in 2017. He is currently working toward the Ph.D. degree in resilient CPS at the Department of Information and Communication engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, South Korea.

His research interests include security of cyber-physical systems and software defined networking.

Yongsoon Eun (Senior Member, IEEE) received the B.A. degree in mathematics and the B.S. and M.S.E. degrees in control and instrumentation engineering from Seoul National University, Seoul, South Korea, in 1992, 1994, and 1997, respectively, and the Ph.D. degree in electrical engineering and computer science from the University of Michigan, Ann Arbor, MI, USA, in 2003.

From 2003 to 2012, he was a Research Scientist with the Xerox Innovation Group, Webster, NY, USA, where he worked on technologies in the xerographic marking process and production inkjet printers. Since 2012, he has been with the Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, South Korea, and is currently a Professor with the Department of Information and Communication Engineering and also Director of DGIST Resilient Cyber-Physical Systems Research Center. His research interests include control systems with nonlinear sensors and actuators, control of quadrotors, communication network, industry 4.0 production systems, railroad vehicle platooning, and resilient cyber-physical systems.

Kyung-Joon Park (Senior Member, IEEE) received the B.S. and M.S. degrees in electrical engineering, and the Ph.D. degree in electrical engineering and computer science from Seoul National University, Seoul, South Korea, in 1998, 2000, and 2005, respectively.

From 2005 to 2006, he was a Senior Engineer with Samsung Electronics, Suwon, South Korea. From 2006 to 2010, he was a Postdoctoral Research Associate with the Department of Computer Science, University of Illinois, Urbana-Champaign (UIUC), Champaign, IL, USA. He is currently a Professor with the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology, Daegu, South Korea. His research interests include resilient cyber-physical systems and smart production systems.