A Survey on Network Security for Cyber–Physical Systems: From Threats to Resilient Design

Sangjun Kim, Kyung-Joon Park, *Senior Member, IEEE*, and Chenyang Lu, *Fellow, IEEE*

***Abstract*—Cyber-physical systems (CPS) are considered the integration of physical systems in the real world and control software in computing systems. In CPS, the real world and the computing systems are connected through networks with real-time information exchange. The introduction of networking technologies in CPS has substantial advantages in terms of system efficiency, scalability, maintenance, and many more. However, CPS are vulnerable to malicious attackers intruding on the network. The attackers aim to destroy the physical systems with cyber-physical attacks, in which the main objective is to remotely cause malfunctions in physical systems through networks. Therefore, extensive research on cyber-physical security has been conducted to detect cyber-physical attacks and guarantee the stability of the physical systems under cyber-physical attacks. In this survey, we conduct an exhaustive review of the literature on threats to CPS and on resilient CPS design strategies. First, we discuss the structure of CPS, considering physical dynamics, computing systems, and networks. Then, we provide a taxonomy of cyber-physical attacks with three properties (attack space, attack location, and stealthiness). These three attributes represent the requirements for the implementation of cyber-physical attacks. We review existing studies on anomaly-detection strategies against cyber-physical attacks in terms of physics, networks, and machine learning techniques. Moreover, we analyze the impact of typical cyber-physical attacks and the constraints on attack implementations. In particular, for each of the well-established cyber-physical attacks, we present numerical examples that clearly illustrate the time responses of the physical systems with a conventional physics-based anomaly detector. We also review advances in the typical cyber-physical attacks and the detection methods for these attacks. Furthermore, we review resilient CPS designs in the context of control theory, network management, and data-driven technologies. We conclude the survey by identifying future research directions.**

***Index Terms*—Cyber-physical systems, cyber-physical attacks, cyber-physical security, attack detection, attack resiliency.**

Manuscript received 3 March 2021; revised revised 16 September 2021, 13 February 2022, and 3 May 2022; accepted 12 June 2022. Date of publication 30 June 2022; date of current version 23 August 2022. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIT) under Grant NRF-2019R1A2C1088092. *(Corresponding author: Kyung-Joon Park.)*

Sangjun Kim and Kyung-Joon Park are with the Department of Electrical Engineering and Computer Science, Daegu Gyeongbuk Institute of Science and Technology, Daegu 42988, South Korea (e-mail: sjkim@dgist.ac.kr; kjp@dgist.ac.kr).

Chenyang Lu is with the Department of Computer Science and Engineering, Washington University in St. Louis, St. Louis, MO 63130 USA (e-mail: lu@cse.wustl.edu).

Digital Object Identifier 10.1109/COMST.2022.3187531

I. Introduction

CYBER-PHYSICAL systems (CPS) are basically networked control systems (NCS) in the sense that multiple physical systems in the real world and control software in the cyber world are connected through networks. Wired and wireless networks create feedback control loops between the physical and cyber worlds [1]–[4]. With the advancements in communication technologies, real-time information exchange between the real world and the cyberspace has become feasible. This improved connectivity allows for the control software to provide more precise and intelligent services to physical systems in real-time [5], [6]. In addition, the wired/wireless networks of CPS reduce the configuration costs and improve flexibility compared to the conventional point-to-point, connection-based control systems [1], [4].

Owing to its advantages, including real-time capability through the development of communication technologies, CPS are emerging as core technologies in various industries, such as smart factories [7], intelligent transportation systems [8], [9], medical systems [10], and smart grids [11]. For example, in an autonomous transportation system, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications are essential technologies for the autonomous driving of vehicles and for traffic congestion control [9]. Bidirectional communication in a smart grid provides intelligent services to customers, including power demand prediction, power production control, and fault-tolerant power transmission.

However, improved connectivity in networks renders CPS vulnerable to external threats [12], [13]. To damage the traditional embedded control systems, a malicious user must physically access the system. However, CPS can be remotely accessed by a malicious attacker through networks. In particular, a malfunction in CPS for social infrastructure can cause economic loss, equipment damage, and serious personal injury. In 2015, the Blackenergy attack, which was launched on the Ukrainian power grid through an unsecured network, caused a massive blackout [14]. In 2014, a malicious attacker invaded a German steel mill company's network and inactivated the blast furnace on the production line, causing major damage to the facility [15].

The purpose of this survey is to provide a *comprehensive review on how cyber-physical attacks damage physical systems through networks* and *how we can design resilient CPS against cyber-physical attacks*. We analyze the time response of physical systems with respect to cyber-physical attacks by using

1553-877X © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

TABLE I
COMPARISON OF OUR SURVEY WITH EXISTING ONES

References	Attack location (Sec. V)			Attack detection (Sec. VI)		Attack analysis (Sec. VII)		Prevention and resilient strategy (Sec. VIII)		
	Sensor	Controller	Combined	Detection methods†	Real-time detection	Physical impact	Quantitative analysis	Secure control	Network management	Data-driven strategy
[12]	●	●	●	P	●	●	●	-	-	-
[23]	●	●	-	P	●	●	●	-	-	-
[28]	●	●	●	N, P	-	●	-	●	-	-
[29]	●	●	●	P	-	●	-	●	-	-
[30]	●	-	-	P	-	◐	-	-	●	-
[31]	-	●	-	N	-	-	-	●	-	-
[32]	●	●	-	N	-	◐	-	●	●	-
[33]	●	●	●	N, P	-	●	-	●	●	-
[34]	●	●	-	N, P	-	◐	-	●	-	-
[35]	●	●	●	N	-	-	-	●	●	◐
Our survey	●	●	●	P	●	●	●	●	●	●

● = Applicable; ◐ = Not explicitly stated or exhibits ambiguity;
† P = Physical knowledge based detection (by sensor measurements), N = Network knowledge based detection (by network intrusion detection systems)

control theory as an analytical tool. The contributions of this survey are as follows:

- From a networking viewpoint, we provide a comprehensive survey of cyber-physical attacks up to date.
- We show the time response of the physical system with respect to each cyber-physical attack and check the stability of the physical system by systematically implementing *illustrative numerical examples*.
- We provide resilient CPS design strategies from the perspectives of networking, control theory, and data-driven approaches.

II. PRELIMINARIES

A. Cyber-Physical Attacks and Cyber-Physical Security

A *cyber-physical attack* is defined as the exploitation of CPS by inflicting harm on networks to connect computing and physical systems [5], [12]. The objective of cyber-physical attacks is to remotely cause malfunctions in physical systems via networks [16]. To implement cyber-physical attacks, intruders utilize methods like malware injection and man-in-the-middle (MITM) attacks [17] using network protocol vulnerabilities, network resource exhaustion, and intended communication disruptions [5], [18].

Recently, control-theoretical stealthy cyber-physical attacks are reported: the replay attack [19], the zero-dynamics attack (ZDA) [20], the pole-dynamics attack (PDA) [21], and the covert attack [22]. These stealthy attacks elaborately manipulate control input signals and sensor measurements on the network. Because conventional computing systems are unable to detect these stealthy attacks [12], CPS require security in various layers, such as computation, network-centric, as well as control-theoretic. Conventional cyber security mainly focuses on information security such as eavesdropping prevention and encryption [23].

Cyber-physical security introduces the dimension of physical dynamics in addition to conventional cyber security, and mainly focuses on the stability guarantee for physical systems [12], [16]. In CPS, eavesdropping attacks do not affect the behavior of the physical system [19]. In addition, encryption techniques can be used as a major countermeasure against cyber attacks. However, processing delays caused by encryption and decryption interfere with real-time interaction of CPS. These delays degrade the control performance of the physical system [24]. Because of the real-time properties of physical dynamics, cyber-physical security needs to consider both physical and network aspects.

There are two aspects that cyber-physical security covers: passive security and active security [25]. Passive security prevents access by attackers through encryption and strict network access control policies. Active security recovers CPS when cyber-physical attacks are launched. In active security, various recovery strategies, including link-failover and attacker isolation [26] are adopted to guarantee the resiliency of CPS [27]. While many studies have been conducted on passive security in traditional cyber security, there have been insufficient studies on active security that consider both the dynamics of physical systems and the resiliency of networks.

B. Related Surveys

There are a number of surveys on security for CPS. Most of the studies are conducted in the context of control theory or network fields for the Internet of Things (IoT). Table I shows a comparison between existing surveys and ours. Most surveys listed in Table I present the contents of attack location, attack detection, attack analysis, and prevention and resilience strategy, which correspond to Sections V, VI, VII, and VIII of our survey, respectively. These four items are common contents that conventional CPS security surveys mainly cover. We classify these four contents into three categories: cyber-physical attacks, detection strategies, and prevention and resilient strategies. In this section, we clarify the distinctions between our survey and the existing cyber-physical security surveys in Table I. In surveys based on control theory [12], [23], [28]–[30], the main contents analyze cyber-physical attacks from the perspectives of time-series and mathematics. Furthermore, detection methods and resilient control techniques are dependent only on physical system dynamics in the control-theoretical surveys.

In addition, surveys based on the network knowledge [31]–[34] cover analysis, detection, and prevention for

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

the conventional IT security in the IoT environment. Therefore, the existing surveys in the conventional network area cannot properly reflect the physical behavior by the cyber-physical attacks in the real-time perspective.

Furthermore, both control-theoretical surveys and network-based surveys provide insufficient countermeasures against cyber-physical attacks; even the countermeasures mentioned in those surveys mainly focus on attack prevention only. In contrast to the existing surveys, we provide time response analysis for cyber-physical attacks using control theory and anomaly detection. Furthermore, our survey provides adequate resilient CPS designs to handle cyber-physical attacks from the perspectives of control, network, and machine learning.

1) Cyber-Physical Attacks (Sections V and VII): From the viewpoint of cyber-physical attacks, control-theoretical surveys mostly consider cyber-physical attacks as mathematical modifications of signals of feedback control systems. Thus, the control-theoretical surveys mainly focus on attack location and mathematical formulation of cyber-physical attacks. Surveys in [23], [28] classify attack location that refers to the type of data modified by the attacker: sensor attack and controller attack. Surveys in [12], [29] analyze several cyber-physical attack strategies against CPS in the context of control theory and quantitatively show the impact of the cyber-physical attacks on the physical systems in the time domain. Among the cyber-physical attacks, stealthy cyber-physical attacks that avoid control-theoretical detectors, are the most important issues in the control theory field. Control-theoretical surveys [12], [23], [29] provide a concept of stealthiness property and several stealthy cyber-physical attacks. In particular, authors in [23] provide realistic examples of a stealthy cyber-physical attack by implementing a testbed.

In contrast, the network-knowledge based surveys regard cyber-physical attacks as information leakage and fault data injection, and provide insufficient information about the attack impact on the physical systems. For the fault data injection attacks, although the network-knowledge-based surveys do not strictly classify the attack types, such as sensor attack and controller attack, the sensor measurement manipulation and the malicious command injection can be considered as the sensor attack and the controller attack, respectively. Furthermore, the network-knowledge based surveys have limited information on the physical impact of the cyber-physical attacks and do not provide quantitative analysis of the attacks. Network attacks for IoT are presented in [31] considering passive network attacks such as leakage of privacy data. The disruption of the physical systems is out of scope in [31]. Furthermore, most of the cyber-physical attacks presented in [34] are information leakages, such as key stroke inferences, in conventional IoT systems. Extensive vulnerabilities and methodologies that disrupt networked systems are analyzed in terms of device, network, and software in [32], [33], where the authors present a realistic threat of cyber-physical attacks. However, similar to [31], the surveys [32], [33] do not provide specific impacts on physical systems of the threats.

In our survey, we classify the cyber-physical attacks into the sensor attack, the controller attack, and the combined attack. Furthermore, we provide a mathematical representation of these cyber-physical attacks and constraints for the attack generation from the viewpoint of the attacker. By providing numerical simulations of the cyber-physical attacks, we show the physical impact and stealthiness of the cyber-physical attacks in the time domain.

2) Detection Strategies (Section VI): Detection strategies are common in both control-theoretical surveys and network-knowledge based surveys for cyber-physical security. However, the surveys in these research fields have different detection approaches for cyber-physical attacks.

The control-theoretical surveys provide limited attack detection techniques by sensor measurements and control input signals on the feedback control systems. The control-theoretical detection approaches provide a real-time attack detection concept that includes an analysis of the impact of cyber-physical attacks on physical systems. Surveys [12], [23] provide a basic detection technique for anomaly detection, based on the deviation between sensor measurements from the networks and sensor measurement predictions based on control system knowledge. In particular, the authors in [23] highlight the real-time attack detection with the testbed implementation. A survey [29] provides various cyber-physical attack detection methods against certain cyber-physical attacks. However, these detection methods are based on the knowledge on the control system. Furthermore, several machine-learning techniques, based on sensor measurements and control input signals are introduced in [28], [30], which support classification of cyber-physical attack types and locations.

In general, network intrusion detection systems (IDS) are considered in network-knowledge based surveys as a basic attack detection technique. In [31], [34], a comprehensive survey on IDS is presented in the IoT environment. Furthermore, a control-theoretical survey [28] introduces IDS as the cyber-physical attack detection method. A survey [32] presents not only IDS, but also various network-layer attack detection techniques such as honeypot, and vulnerability assessments for known network vulnerabilities. Overall, the conventional network IDS and other detection techniques introduced in network-knowledge-based surveys only focus on the detectability of cyber-physical attacks, and do not address real-time detection.

Different from existing surveys, our survey provides details of various control-theoretical cyber-physical attack detection techniques. The provided detection techniques support real-time attack detection.

3) Prevention and Resilient Strategies (Section VIII): In contrast to attack analysis and detection, there are insufficient surveys for attack prevention and resilience with consideration of physical system dynamics in both control-theoretical surveys and network-knowledge based surveys. We focus on secure control, network management, and data-driven strategies for attack prevention and resilience, where secure control includes data encryption because the sensor measurements and control input signals can be encrypted in the packet transmission processes.

Control-theoretical surveys mainly consider encryption and various secure control algorithms from the viewpoint of attack prevention. Surveys [28], [29] focus on attack prevention by

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

data encryption, and present various data encryption methods. A survey [30] provides resilient design methods against system faults on the ship's power systems. The authors in [30] present various fault isolation methods to prevent propagation faults and system reconfiguration methods to ensure critical functions related to the safety of the networks.

Along with control-theoretical surveys, all network-knowledge-based surveys mainly consider encryption, but present various attack prevention and handling techniques from the viewpoints of network management and data-driven methods. Basically, surveys [31], [32] introduce authentication methods and secure protocols with lightweight cryptography techniques for cyber-physical attack prevention. The surveys [32], [33] consider strict firewall methods as a basic defense method against cyber-physical attacks. A survey [33] introduces other network management techniques, such as distance-limited communication, to prevent remote control of the attacker. In [35], the authors mention advanced IDS with machine-learning techniques and secure routing to mitigate cyber-physical attacks.

Other existing surveys focus on prevention against cyber-physical attacks, whereas our survey provides various resilient CPS designs in three aspects: secure control, network management, and data-driven strategy. These resilient CPS designs recover the physical impact of the cyber-physical attacks, and ensure the safe operation of physical systems under cyber-physical attacks. Furthermore, we provide two resilient CPS examples to enhance the understanding of the safe operation of resilient CPS under cyber-physical attacks.

C. Structure of This Paper

The structure of the paper and relations between sections are illustrated in Fig. 1. Section III presents basics on control theory, which serves as a mathematical tool for analyzing the impact of attacks on physical systems. Section IV introduces the hierarchical CPS structure and presents the physical system model used for analyzing the physical impact of cyber-physical attacks. Section V describes the taxonomy of cyber-physical attacks according to the cyber-physical attack space, attack location, and stealthiness. Section VI presents general cyber-physical attack detection strategies that utilize physical dynamics and network knowledge. Section VII discusses the physical impact of typical cyber-physical attacks by presenting numerical examples, requirements for the attacker, and detection strategies. Section VIII presents resilient CPS structures that can withstand cyber-physical attacks and unexpected system faults. In Section VIII, we classify CPS structures into secure control, network management, and data-driven strategies. Section IX offers potential research directions. Section X summarizes this survey. We conclude this survey in Section XI.

It should be noted that we focus on *attacks that can directly damage the physical dynamics.* Therefore, conventional network attacks such as eavesdropping and cryptanalysis are outside the scope of our survey. In addition, we mainly consider the attacks launched on the network. Thus, software malfunctions by malware and destruction of the physical system by intentional physical impact are not covered.

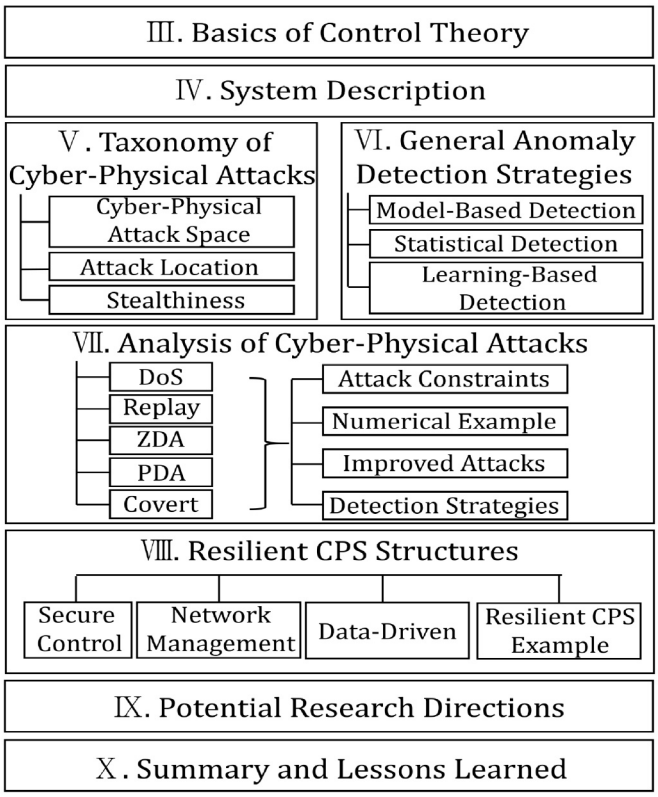


Fig. 1. Structure and relations of the sections.

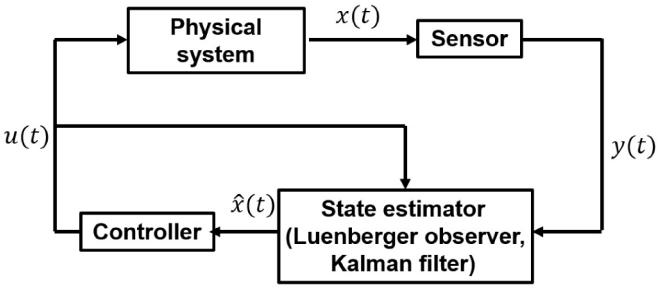


Fig. 2. Block diagram of a feedback control loop.

III. Basics on Control Theory

One of the main functions of CPS is to enable multiple physical systems to be operated as intended by the user. Therefore, the main objective of a cyber-physical attack is to disturb the temporal behavior of the physical systems. We adopt control theory to analyze the time response of physical systems in CPS with respect to cyber-physical attacks. In this section, we provide basic concepts and terminologies in control theory used in this survey.

As illustrated in Fig. 2, a control system consists of four components: a physical system, a controller, a sensor, and a state estimator. A physical system is an object having physical quantities. The sensor measures certain physical quantities of the physical system. The state estimator calculates all the physical quantities of the physical system from the sensor measurements based on a mathematical model of the physical system. The controller generates a control input signal to

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

TABLE II
MATHEMATICAL NOTATION FOR CONTROL THEORY

Symbol	Description
t	Time instant in continuous-time domain
k	Time step in discrete-time domain
x	Physical state
u	Control input signal
y	Sensor measurement
A	System matrix
B	Input matrix
C	Output matrix
D	Feed-forward matrix
$\hat{x}$	State estimation
L	Luenburger observer gain
K	Feedback controller gain
y_r	Reference signal
A_d	System matrix in discrete time domain
B_d	Input matrix in discrete time domain
C_d	Output matrix in discrete time domain
$\tilde{y}$	Modified sensor measurement
y^a	Sensor attack signal
$\tilde{u}$	Modified control input signal
u^a	Controller attack signal
r	Residual
Γ	Attack detection alarm

operate the physical system as intended by the user. The feedback control loop with these four components can generate the proper control input signal to operate the physical system. The mathematical notation for control theory in the paper is given in Table II.

A. State of a Physical System

A physical system has multiple time-varying physical quantities. For example, a vehicle has velocity, acceleration, steering angle, steering angular speed, engine temperature, and others. In control theory, the *state* corresponds to a physical process represented by a vector $x(t) \in \mathbb{R}^n$, whose dimension *n* is the number of the physical quantities. Each physical quantity is represented by each element of the state vector $x(t)$, which is called a state variable. The goal of *control* is to render the state $x(t)$ remain as intended by the user with proper control input signals.

B. System Models

A physical system can have multiple control input signals $u(t)$ handling the state $x(t)$, and sensor measurements $y(t)$, by sensors attached to the physical system. From the inputs-outputs relationship, the physical system is mathematically represented by a differential equation form. In addition, the mathematical model of the physical system provides characteristics of the physical system in terms of pole and zero dynamics and is used to design the state estimator that estimates internal physical quantities that are not directly measured by sensors.

1) System Inputs and Outputs: Control input signals $u(t)$ drive the physical system. Well-designed input signals force the state $x(t)$ as intended by the user. As a result of the control input signals, the physical system generates sensor measurements, $y(t)$, which represent a measurement of the state $x(t)$ by sensors attached to the physical system. However, due to the limitation of sensor attachment, it is not always possible to measure all the state variables. For example, a vehicle system has the following control inputs: the accelerator pedal and the steering wheel change the speed and the steering angle of the vehicle, respectively. The dashboard of the vehicle provides a speed measurement to the driver. Moreover, the eyes of the driver are a kind of sensor that measures the steering angle of the vehicle. However, a measurement of the engine temperature is not available because of the limitation of sensor attachment.

2) Single-Input Single-Output (SISO) Systems: Many physical systems are multi-input multi-output (MIMO) systems, in which the control input signal $u(t)$ and the sensor measurement $y(t)$ are represented mathematically as vectors. However, the MIMO model is inadequate for providing a proper intuition because of the complex relationship between the state $x(t)$, the input signal $u(t)$, and the sensor measurement $y(t)$. In this survey, we consider the physical system to be a single-input single-output (SISO) system with one control input signal $u(t)$ and one sensor measurement $y(t)$.

The control input signal $u(t)$ and the sensor measurement $y(t)$ have scalar values in the SISO model. Though the SISO model does not reflect all possible physical behaviors of CPS, its simplicity provides a valuable intuitive description of the temporal behavior of the state $x(t)$ and the sensor measurement $y(t)$. In the vehicle example, one possible SISO model is to consider one-dimensional motion, where the control input signal $u(t)$ is the acceleration and the sensor measurement is the velocity. Though this SISO model does not reflect the steering of the vehicle as an input at the same time, it provides a clear relation between the time-varying acceleration and the vehicle speed.

3) Linear Time-Invariant (LTI) Systems: A linear system is a mathematical model of a system in which the behavior of the state $x(t)$ is expressed by a linear operator. The linear system satisfies both additivity and homogeneity properties between the physical state $x(t)$ and the control input signal $u(t)$. If the state $x(t)$ does not satisfy either additivity or homogeneity, it is called a *nonlinear system*.

A time-invariant system is defined as a system whose properties do not change with time. In other words, the state $x(t)$ has no direct dependency on time, but only an indirect dependency through $u(t)$, or mathematically $x(t) = f(u(t))$ for a specific function $f(\cdot)$.

Clearly, most physical systems have nonlinear operating characteristics. However, the nonlinear system is extremely hard to analyze in general. Consequently, in most control applications, nonlinear systems are approximated by linear systems. The linearization has distinctive advantages for describing the properties of the system in an intuitive manner and formulating the control problem in a feasible form. In particular, the LTI system model has distinctive advantages

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

for analyzing the impact of cyber-physical attacks. The time responses of the state $x(t)$ by legitimate control and cyber-physical attacks at a certain instant can be separately considered by virtue of the linearity and time-invariant property of the LTI model. Therefore, in this survey, we consider the physical system as a linear time-invariant (LTI) system to analyze the physical impact of various cyber-physical attacks in a proper manner.

C. Mathematical Description of Systems

The motion of a physical system can be represented in a mathematical form. Two typical mathematical models for describing a physical system are mainly used in control theory: the transfer function and the state-space equation.

1) Transfer Function: In control theory, a transfer function is used to describe a physical system using only the control input signal $u(t)$ and the sensor measurement $y(t)$. The input-output convolution relationship in the time domain can be described as follows:

$$y(t) = \int_{-\infty}^{\infty} g(\tau - t)u(\tau)d\tau, \tag{1}$$

where $g(t)$ is the impulse response of the system, defined as the output of a physical system when presented with an impulse input. The transfer function is the Laplace transform of the impulse response $g(t)$, which can be described as the following rational function in the s-domain:

$$G(s) = \frac{N(s)}{D(s)}, \tag{2}$$

where $G(s)$ is the transfer function, $N(s)$ and $D(s)$ are the numerator and the denominator of the transfer function, respectively. We can use a transfer function to analyze the transient behavior and the stability of the physical system by investigating the roots of $D(s)$ and $N(s)$; the roots of the polynomial $N(s)$ are known as the zeros of the physical system, and the roots of the polynomial $D(s)$ are known as the poles. However, the transfer function analysis in the s-domain cannot consider state variables that are not measured by the sensors.

2) State-Space Equations: A state-space representation is a set of differential equations to represent multiple state variables in control theory. A state space representation can be described as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{3}$$

where $A \in \mathbb{R}^{n\times n}$ is the system matrix, $x(t) \in \mathbb{R}^n$ is the state vector of the physical system, $u(t) \in \mathbb{R}$ is the control input, $y(t) \in \mathbb{R}$ is the system output, $B \in \mathbb{R}^n$ is the input matrix, $C \in \mathbb{R}^{1\times n}$ is the output matrix, and $D \in \mathbb{R}$ is the feed-forward matrix. Note that here we assume the SISO LTI system. Most control theoretical studies do not consider the feed-forward matrix D, and hence the matrix D can be omitted.

In contrast to the transfer function, the state-space analysis can provide an explicit relationship between the control input $u(t)$ and the system output $y(t)$ in the time domain. In addition, the state-space analysis considers the internal states that are not directly measured by sensors. However, compared to the transfer function form, the state-space representation requires more operations to express the characteristics of the physical system. For example, the poles are represented as the eigenvalues of the system matrix A, and the zeros are represented by taking the Byrnes-Isidori normal form [36] of the state-space form (3).

D. Characteristics of Systems

Using mathematical system models, we can analyze the physical system and design a proper controller. As already mentioned above, we can investigate the transient behavior as well as the stability of the physical system by looking into the zeros and the poles of the system. In addition, we can design a proper controller by using the mathematical system models.

1) Zero and Pole Dynamics: Zeros of a system are the roots of the numerator polynomial $N(s)$ of the transfer function (2). The *zero dynamics* of a system corresponds to the behavior of the system output $y(t) = 0$ for a specific pair of the system input $u(t)$ and the initial condition $x(0)$. In other words, for a special pair of ($u(t)$, $x(0)$), the system output $y(t)$ will become zero, and this behavior is called the zero dynamics.

The poles of a system are the roots of the denominator polynomial $D(s)$ of the transfer function (2), which determines the stability of the system and how well it performs. A pole on the right-half plane (i.e., the pole that has a positive real-part) intrinsically destabilizes the physical system, which can be observed by the divergence of the output sensor measurement. Furthermore, the imaginary part of a pole determines the oscillation property of the system output $y(t)$.

2) Stability: The stability condition of the physical system is defined as a certain range of the state $x(t)$ that is determined by the specifics of the physical system. For example, the vehicle has a certain state boundary, such as the speed limit or the engine temperature, and the vehicle becomes unstable if any of the state variables exceeds the state boundary. If the state $x(t)$ satisfies the stability condition for all time t, the system is called *stable*. In general, divergence of the state $x(t)$ means instability of the physical system.

Stability can be studied by investigating the poles of the physical system. If any physical system has at least one pole in the right-half plane in the s-domain (i.e., the real part of the pole is positive), the physical system is unstable. Thus, the poles in the right-half plane are called *unstable poles*. Likewise, the zeros on the right-half plane are called *unstable zeros*.

3) Controllability: Controllability is defined as the ability of a control system to reach a definite state from a fixed initial state in a finite amount of time. If a physical system does not satisfy the controllability condition, it is impossible to design a controller. The controllability of a system can be determined by the system matrix A and the control input matrix B.

4) Observability: Observability is a measure of how well the internal states of a system can be inferred from knowledge of its external outputs. If a physical system does not satisfy the observability condition, it is impossible to design a state estimator. The observability of a system can be determined by the system matrix A and the output matrix C.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

E. State Estimator

A physical system has multiple physical quantities. However, because of the practical limitations of the sensor measurements, it is formidable to measure all the state variables of the physical system. To control all the physical quantities with certain measured state variables, we should estimate the other state variables by using a mathematical system model. A state estimator is a mathematical algorithm that can estimate the physical state $x(t)$ based on the sensor measurement $y(t)$ if the physical system satisfies the observability condition.

The Luenburger observer and the Kalman filter are the most common state estimators. In this survey, we mainly consider the Luenburger observer as a state estimator. The Luenburger observer can be described as follows:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L(y(t) - C\hat{x}(t)), \tag{4}$$

where $\hat{x}(t)$ is state estimation and $L \in \mathbb{R}^n$ is the Luenburger observer gain that should be determined for each physical system. As illustrated in equation (4), state estimation requires the physical system model, the control input signal $u(t)$, and the sensor measurement $y(t)$. When there is a difference between the state $x(t)$ in the real physical system and state estimation $\hat{x}(t)$, this difference appears as an error term $y(t) - C\hat{x}(t)$ in the state estimator (4), and the Luenburger observer gain L compensates for it. If the state estimator has a precise model of the physical system and the Luenburger observer gain L is properly selected, the state estimator can accurately estimate the actual state $x(t)$ over time.

F. Controller

The goal of control is to enforce the user's intended state $x(t)$ by injecting a control input signal $u(t)$. To accomplish this, the controller must generate an appropriate control input signal $u(t)$. To design the controller, the mathematical system model must satisfy the controllability condition.

In this survey, we mainly consider a state-feedback controller generating the control input signal $u(t)$ by state estimation $\hat{x}(t)$. The state-feedback controller is represented as follows:

$$u(t) = Py_r(t) - K\hat{x}(t), \tag{5}$$

where P is a scalar gain, $y_r(t)$ is a reference signal, and $K \in \mathbb{R}^{1\times n}$ is a state feedback controller gain. If the controller gain K is properly selected, the sensor measurement $y(t)$ arrives at the reference signal $y_r(t)$ in a finite amount of time.

IV. System Description

This section provides an overview of CPS in terms of physical systems, networks, and computing systems. In addition to the CPS architecture, we provide the overall temporal behavior of CPS.

A. Hierarchical CPS Structure

We consider a hierarchical CPS architecture with the following three layers: the physical layer, the network layer, and the application layer. Compared to CPS without any hierarchical structure, hierarchical CPS have substantial benefits in maintaining systems and handling novel security issues. We can apply this hierarchical CPS architecture to realistic CPS, such as intelligent transport systems, communication-based train control (CBTC) systems, smart production systems, and smart grids [8].

1) Physical Layer: The physical layer represents the physical systems in the real world. In the physical layer, the physical systems are connected through networks, and sensors attached to the physical systems periodically transmit measurements to a centralized computing system. In addition, the physical systems perform actuation according to a control input signal received through the network from the computing system of the application layer. In general, because of the power limitations on the physical systems, task performance with complex computations in the application layer depends on the physical layer.

Unmanned aerial vehicles (UAV) and autonomous trains are examples of physical-layer components for intelligent transport systems, where they operate according to physical dynamics in the real world [37], [38]. In the industrial area, production facilities such as machines and conveyor belts are examples of physical systems. Similarly, electric equipment that physically transmits electricity, power plants, and distribution stations are part of the physical layer of a smart grid.

2) Network Layer: The network layer represents the feedback loop of CPS and supports interactions between the physical layer and the application layer. The network layer connects the physical layer and the application layer through a wired or wireless network and is responsible for data exchange between these two layers [4]. Because the network in CPS must provide services to the physical systems operating in a continuous-time domain, the network must guarantee data transmission in real-time [10], [39].

For vehicle-to-everything (V2X) communications, the 802.11p standard [40] and 5G communication protocols are proposed [9]. In addition, CBTC systems adopt LTE-R [41] or 802.11-based protocols for train-ground station communications [8]. In the industrial domain, wired networks such as Modbus [42] or the S7Comm protocol [43], and wireless sensor networks (adopting, for example, WirelessHART [44] or Zigbee [45]) carry out the data exchanges between production facilities [46]. In addition, the distributed network protocol (DNP3) [47] and IEC 61850 [48] networks connect electric equipment and exchange data in a smart grid.

3) Application Layer: The application layer represents the computing systems in the cyber-world. The application layer supervises the physical systems and provides functions to CPS users. The computing systems estimate the overall state of the entire CPS based on sensor measurements received from multiple physical systems. Then, the computing systems determine the next state of the physical system and transmit the control input signal for the physical system to execute commands in the applications.

In transport systems, a traffic management center estimates the traffic conditions based on data received from vehicles and

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

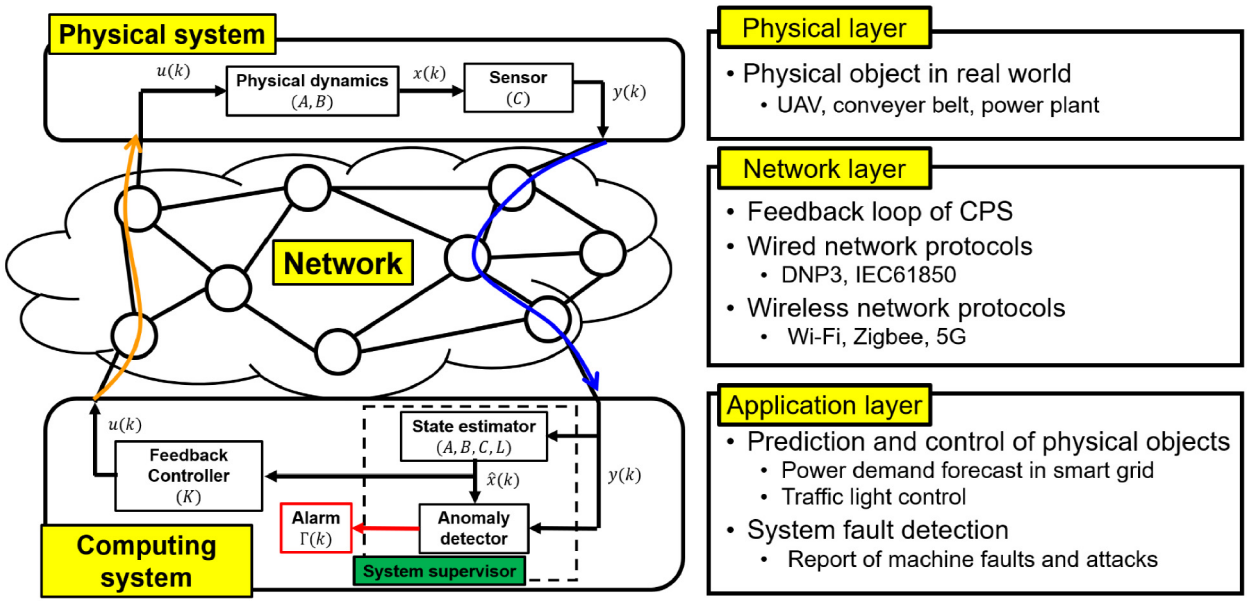


Fig. 3. Structure of a networked control system.

provides traffic flow and congestion control services to the traffic infrastructure, such as the traffic lights at intersections [49]. In manufacturing systems, the supervisory control and data acquisition (SCADA) systems are typically the most centralized application systems. The SCADA systems predict production throughput, establish production schedules, and report production machine fault information to an administrator. In addition, the SCADA systems of a smart grid can forecast power demands, control power generation, and provide fault-tolerant power transmission by utilizing sensor data from the infrastructure.

B. Networked Control Systems (NCS)

CPS are the integration of physical systems, networks, computing hardware, and various application software in a broad sense. The primary focus of our survey is to specifically provide the impact of cyber-physical threats and anomaly detection techniques, based on control theory. Therefore, we introduce the concept of NCS widely used in the control theory field. NCS are defined as feedback control systems, where physical systems and feedback controllers are connected through networks. NCS have a more concrete concept than CPS, which focuses on the behavior of physical systems from a control-theoretical viewpoint.

CPS are basically NCS from a control-theoretic viewpoint. NCS consist of three main components: the physical system, the network system, and the computing system. In the hierarchical structure of CPS, each of these three components corresponds to the physical layer, the network layer, and the application layer, respectively. In this survey, we mainly use SISO-LTI systems as illustrative examples. However, we also acknowledge studies that handle more complex systems. The computing system has a state estimator, a feedback controller, and an anomaly detector, as shown in Fig. 3.

1) Physical System: The physical system implements physical processes via system dynamics in a continuous-time domain. The dynamics of the physical system can be described as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t), \end{aligned} \tag{6}$$

where $A \in \mathbb{R}^{n\times n}$ is the system matrix, $x(t) \in \mathbb{R}^n$ is the state of the physical system, $B \in \mathbb{R}^n$ is the input matrix, $C \in \mathbb{R}^{1\times n}$ is the output matrix, $u(t) \in \mathbb{R}$ is the control input, and $y(t) \in \mathbb{R}$ is the sensor output. We assume that the pair (A, B) is controllable and the pair (A, C) is observable. In addition, we assume that the matrix A has eigenvalues whose real parts are positive. The eigenvalues with a positive real part of the matrix A in the system (6) refer to the unstable poles of the physical system [50].

2) Computing System: The sensor measurement y and the control input signal u in NCS are periodically sampled and transmitted in the discrete-time domain. To control the physical system with sampled sensor measurements, the state estimator and the feedback controller in the computing system are designed with discretized system dynamics. The discretized system model with the zero-order hold (ZOH) can be described as follows:

$$\begin{aligned} x(k+1) &= A_d x(k) + B_d u(k) \\ y(k) &= C_d x(k), \end{aligned} \tag{7}$$

where $A_d \in \mathbb{R}^{n\times n}$, $B_d \in \mathbb{R}^n$, and $C_d \in \mathbb{R}^{1\times n}$ are discretized system, input, and output matrices via ZOH, respectively.

Because the sensor measurement, $y(k)$, does not always capture all the states of the physical system, the computing system can use various state estimators (such as the Luenberger observer and the Kalman filter) to obtain a state estimate $\hat{x}(k)$ if the pair (A, C) is observable. In this survey, we consider

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

TABLE III
CATEGORIZATION OF COMMON CYBER-PHYSICAL ATTACKS

Cyber-physical attacks	Attack space			Attack location		Stealthiness
	System knowledge	Disclosure resources	Disruption resources	Sensor	Controller	
Eavesdropping attack	None	✓	-	-	-	✓
Denial of service (DoS) [18]	None	-	✓	-	-	-
Replay attack [19]	None	✓	✓	✓	✓	✓
Zero-dynamics attack (ZDA) [20]	High	-	✓	-	✓	✓
Robust zero-dynamics attack (RZDA) [52]	Intermediate	✓	✓	-	✓	✓
Pole-dynamics attack (PDA) [21]	High	-	✓	✓	-	✓
Robust pole-dynamics attack (RPDA) [21]	Intermediate	✓	✓	✓	-	✓
Covert attack [22], [53]	High	✓	✓	✓	✓	✓

the Luenberger observer described as follows:

$$\hat{x}(k+1) = A_d\hat{x}(k) + B_d u(k) + L(y(k) - C_d\hat{x}(k))$$
$$u(k) = Py_r(k) - K\hat{x}(k), \tag{8}$$

where $\hat{x}(k) \in \mathbb{R}^n$ is the state estimation, $L \in \mathbb{R}^n$ is the observer gain, $K \in \mathbb{R}^{1\times n}$ is the controller gain, P is the scalar gain, and $y_r(k) \in \mathbb{R}$ is the reference signal. We assume that L and K are properly selected, and that $A_d - LC_d$ and $A_d - B_dK$ are Schur matrices to stabilize physical systems (6). The scalar gain P is selected to satisfy $C_d(I_n - A_d + B_dK)^{-1}BP = 1$ for sensor measurements to asymptotically trace the reference signal $y_r(k)$, where I_n is the identity matrix with n dimensions.

Based on the sensor measurement $y(k)$ and the state estimate $\hat{x}(k)$, an anomaly detector checks whether the physical system is operating normally or is under cyber-physical attack. If an attack is detected, the computing system handles the countermeasures, such as changing the control policy to recover the physical system from the impact of the attack. More details are covered in Sections VI and VII.

3) Network: The networks construct feedback loops between the physical systems and the computing systems. The feedback loops are responsible for exchanging the control input signals and the sensor measurements [37], [51]. Because of the inherent uncertainty of the network, various network faults can occur, such as transmission delays by an unreliable communication channel or packet collisions. In addition, as shown in Fig. 3, cyber-physical attacks implemented through manipulation of control input signals or sensor measurements can be launched against the network. These cyber-physical attacks can drive the physical system into an irreparable state. More details on cyber-physical attacks are covered in Section V and Section VII.

V. TAXONOMY OF CYBER-PHYSICAL ATTACKS

Conventional cyber attacks have been typically classified according to network layers and protocols. However, because the purpose of a cyber-physical attack by a malicious attacker on the network is to destabilize a physical system, the classification of an attack needs to be extended to the dimension of the physical system.

Table III categorizes typical cyber-physical attacks: the eavesdropping attack, the denial of service (DoS) attack, the replay attack, the (robust) zero-dynamics attack, the (robust) pole-dynamics attack, and the covert attack. We explain and analyze each of these attacks in detail in Section VII.

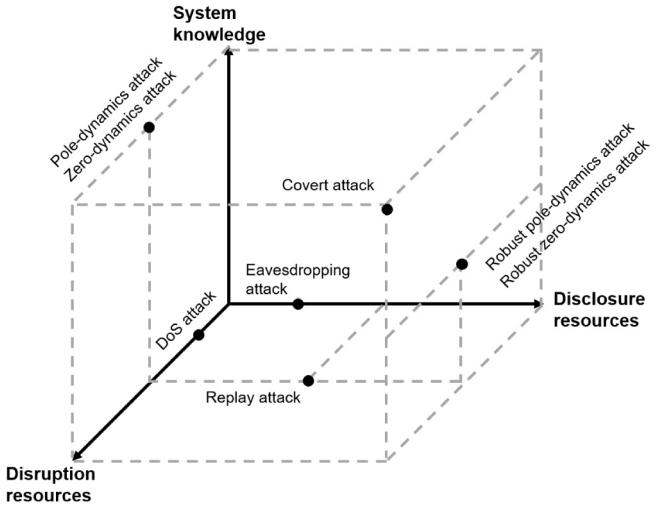


Fig. 4. Cyber-physical attack space.

Cyber-physical attacks can be classified by three perspectives: attack space, attack location, and stealthiness. The attack space represents the requirements for designing an attack and the physical impact of the attack. The attack location refers to the type of data the attacker has to handle on the network. Stealthiness indicates whether the cyber-physical attack can be detected by a conventional anomaly detector in a computing system or not.

In this section, we provide details on cyber-physical attacks according to attack space, location, and stealthiness.

A. Cyber-Physical Attack Space

The cyber-physical attack space represents the characteristics of cyber-physical attacks [12]. The attack space consists of three dimensions: system knowledge, disclosure, and disruption resources. Fig. 4 illustrates typical cyber-physical attacks in the attack space.

The system knowledge vector is a constraint on the required physical system model accuracy when designing an attack. In the system (6), system knowledge corresponds to the system matrices A, B, and C. The requirement for system knowledge refers to the difficulties that the attacker faces when constructing attack signals because precise knowledge on system matrices A, B, and C is difficult to obtain. In contrast, the more accurate system knowledge the attacker has, the more difficult it is for the anomaly detectors to detect the attack.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

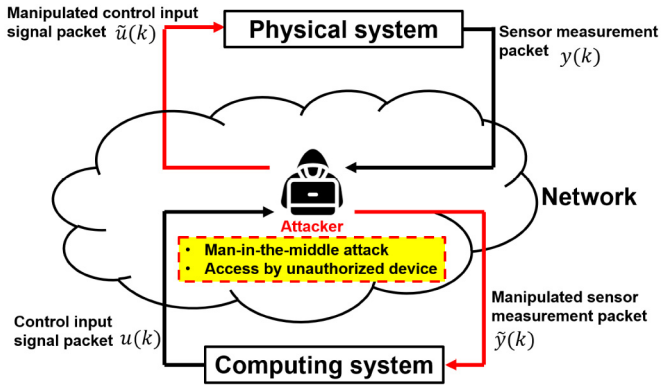


Fig. 5. Man-in-the-middle attack for cyber-physical attack implementation.

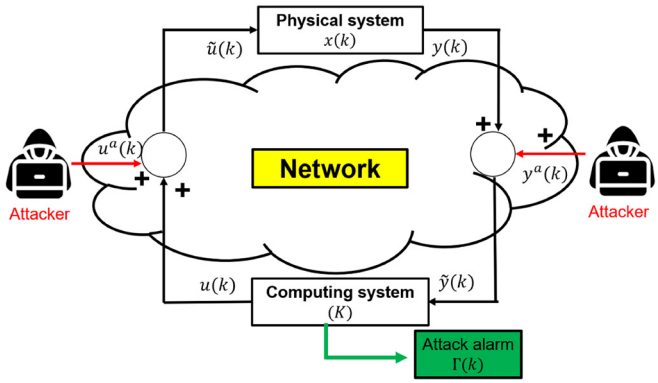


Fig. 6. Cyber-physical attack location.

Disruption resources refer to the level of the physical impact that can be inflicted on a physical process when a cyber-physical attack is launched against the network. Disclosure resources correspond to the amount of information required to generate the attack. In contrast to disruption resources, disclosure resources have no direct effect on the behavior of the physical system.

Consequently, a cyber attack that does not exploit any disruption resources is not considered a cyber-physical attack because it does not directly damage the physical system. For example, an eavesdropping attack is fatal in the traditional cyber-security area because it can directly sniff the packets exchanged on the network. However, it is not considered a cyber-physical attack in the sense that it does not directly affect the operation of the physical system.

B. Attack Location

An attack location in CPS has a different meaning from an attack location in conventional cyber-security. The attack location in conventional cyber-security is defined as the network devices in which the attacker intrudes. However, the attack location in CPS is defined as the type of data manipulated by the attacker. Most cyber-physical attacks, except the DoS, are implemented by the modification of sensor measurements or control input signals. The aim of this manipulation is to destabilize physical systems. Data manipulation of cyber-physical attacks is typically implemented via MITM attacks. By exploiting network protocol vulnerabilities or accessing unauthorized devices, MITM attacks drop legitimate packets and inject false data into the network, as shown in Fig. 5.

As shown in Fig. 6, cyber-physical attacks in linear systems (6) are mathematically expressed as adding an attack signal to the sensor measurement or control input signal in the feedback loop. In Table III and Fig. 6, data-manipulation attacks are further classified into sensor attacks, controller attacks, and combined attacks.

1) Sensor Attack: A lot of sensor attacks targeting smart phones and general IoT devices have already been explored. The results of general sensor attacks include privacy information leakage, task inference, and malfunction of physical systems [54]. Furthermore, the sensor attacks can be injected by various methods, such as electromagnetic ways, optical ways, malware injection, and network packet manipulation [34]. In this survey, we focus on the sensor attacks that affect the disruption of the physical systems and consider that the sensor attacks are only implemented on the networks.

A sensor attack is defined as a modification of a sensor measurement $y(k)$ as follows:

$$\tilde{y}(k) = y(k) + y^a(k), \tag{9}$$

where $\tilde{y}(k)$ is the sensor measurement modified by the attacker, $y(k)$ is the actual sensor measurement, and $y^a(k)$ is the sensor-attack signal added to $y(k)$. When the sensor attack in (9) is launched against the network, the computing system defectively estimates the physical state because the modified value $\tilde{y}(k)$ rather than the actual measurement $y(k)$ is used in the estimation. This defective estimation may cause control failure of the physical system.

Existing studies on cyber-physical security typically focus on sensor attacks [23]. For example, voltage deceiving in power grids [55], [56] is one of the sensor attacks. One of the most common sensor attacks is the pole-dynamics attack (PDA) [21], which adds a sophisticated sensor-attack signal based on the precise physical system model. A more detailed explanation of PDA is presented in Section VII-D.

2) Controller Attack: A controller attack is defined as a modification of the control input signal $u(k)$, which can be represented as follows:

$$\tilde{u}(k) = u(k) + u^a(k), \tag{10}$$

where $\tilde{u}(k)$ is the control input signal modified by the attacker, $u(k)$ is the actual control input signal, and $u^a(k)$ is the controller-attack signal. When the controller attack (10) is launched, the physical system does not operate as intended because of the system's response to the attack signal $u^a(k)$ modified from $u(k)$.

One of the most common controller attacks is the zero-dynamics attack (ZDA), which adds an elaborate adversarial signal to the original control input signal on the network. In addition, control-message hijacking in a CBTC network [16], [57], a power plant [58], and in UAV [59] is an example of a controller attack.

3) Combined Attack: A combined attack is defined as a simultaneous modification of sensor measurements and control input signals. For instance, the replay attack [19] and the

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

covert attack [22] are typical combined attacks. The combined attack needs to hijack both communication channels. In addition, combined attacks incur a higher computation cost than sensor-only attacks or controller-only attacks. Consequently, it is difficult to realize a combined attack. Most combined attacks are more difficult to detect than unidirectional cyber-physical attacks. The covert attack is the most difficult to detect because of the bidirectional modification of both the control input signal and sensor measurement at the same time [53].

C. Stealthiness

To guarantee the stability of the physical system under cyber-physical attacks, detection of the attack should be preceded. Stealthiness, which indicates whether an attack can be detected, is an important property to consider when designing CPS security against specific attacks. In this survey, we assume that cyber-physical attacks are launched through the network by exploiting certain network vulnerabilities. Therefore, it is assumed that the attacks are not detected by conventional IDS, and we consider detectors that utilize physics-based knowledge.

Stealthiness is defined as the property under which an attack is not detected by an anomaly detector designed based on the physical system model. A well-designed attack signal deceives the anomaly detector into realizing a stable state, whereas the actual state of the physical system becomes unstable. Cyber-physical attacks composed of arbitrary signals are mostly detected by an anomaly detector in the computing system, and the physical impact of the attack can be mitigated by resilient control strategies. Therefore, from the viewpoint of the attacker, to destabilize the physical system consistently and avoid detection simultaneously, cyber-physical attacks must sustain stealthiness, the stealthy cyber-physical attack.

To avoid detection by the anomaly detector, stealthy cyber-physical attacks must strictly satisfy stealthiness constraints. The stealthiness constraints can be obtained from the dynamics of the physical system. Therefore, to design an attack with stealthiness, the attacker needs a precise model of the physical system.

For instance, a DoS attack is one of the typical attacks in the network domain, which disturbs the communication between the physical system and the computing system by dropping packets or inducing network delays. From the viewpoint of networking, well-designed DoS attacks by exploiting unknown network vulnerabilities can be *stealthy* because conventional IDS cannot detect the sophisticated attacks. However, no matter what type of DoS attacks are launched on the network, the DoS attack is classified as a non-stealthy attack from the viewpoint of physics because the physics-based detector, which detects the abnormal behavior of the physical systems based on the control input signals and the sensor measurements, can observe the abnormal behavior of physical systems through burst packet drops or significant network delay.

Except the DoS attack, the cyber-physical attacks in Table III have stealthiness when the attacker satisfies the stealthiness constraints. Therefore, to detect these stealthy attacks, new detection strategies need to be developed.

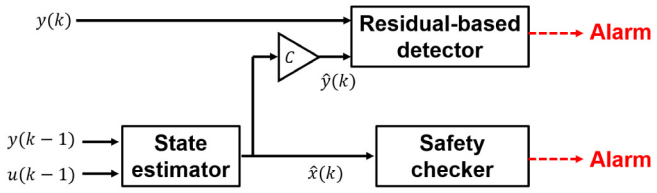


Fig. 7. Physical model-based anomaly detector.

VI. General Anomaly Detection Strategies

To guarantee the safety of CPS under various cyber-physical attacks presented in Section V, attack detection process should be preceded. In this section, we introduce anomaly detection methods for sensor measurements and control input signals. We consider three types of anomaly detection methods: model-based detection with control-theoretical knowledge, statistical detection utilizing the noise property of sensor measurements, and learning-based detection with enormous CPS information. We summarize references for general anomaly detection methods against cyber-physical attacks and system faults in Table IV.

A. Model-Based Detection

Model-based detection is defined as determining an anomaly in a physical system based on the physical state predicted by the state estimator [23]. A state estimator with the physical system model of the computing system not only estimates the physical state, but also predicts the physical state in the next time step. The computing system can expect the sensor measurement $\hat{y}(k+1) = C\hat{x}(k+1)$ in the next time step. There are two types of model-based anomaly detection, i.e., the safety checker and the residual-based detector, as shown in Fig. 7.

1) Safety Checker: The safety checker is the simplest anomaly detection method that can confirm the safety of the physical system. The safety checker determines whether the state of the physical system is within its normal operating range or not. If either the state prediction, $\hat{x}(k)$, or the expected sensor measurement, $\hat{y}(k)$, violates the normal operating range or the allowable sensor measurement range, respectively, the computing system determines the anomaly in the physical system. Because the safety checker only determines whether or not the physical system violates the normal range, the detection speed of the cyber-physical attack is slower than other detectors.

2) Residual-Based Detector: The residual-based detector compares the measurements from the sensor measurement $y(k)$ and the expected sensor measurement $\hat{y}(k)$ received by the state estimator. The residual, $r(k) \in \mathbb{R}$, is defined as follows:

$$r(k) = \|y(k) - C\hat{x}(k)\|. \tag{11}$$

If the physical system model of the state estimator is strictly identical to the real physical system and there are no anomalies in CPS, then the predicted state $\hat{x}(k)$ via the control input $u(k-1)$ converges to the physical response $x(k)$. Thus, the residual $r(k)$ converges to zero in a timely manner under normal operation.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
LIST OF ANOMALY DETECTION METHODS

References	Detector type		Key idea
[60]	Model-based detection	Residual (stateless)	Real-time residual calculation to detect cyber-physical attacks targeting water-level control.
[61]	Model-based detection	Residual (stateless)	Adopt residual-based anomaly detector to real UAV systems with linearized physical model.
[62]	Model-based detection	Residual (stateless)	Calculate residual with nonlinear dynamics of physical system and extented Kalman filter (EKF) for sensor attack detection.
[63]	Model-based detection	Residual (stateful)	Adopt the CUSUM algorithm that calculates residual recursively to reduce false-alarm rate and attack detection time in water-level control system.
[55]	Statistical detection	χ^2 detector	Adopt Euclidean distance to enhance the sensor attack detection performance than conventional χ^2 detector.
[56]	Statistical detection	χ^2 detector	Introduce cosine similarity matching technique to power grid system with the Kalman filter-based state estimation.
[64]	Statistical detection	χ^2 detector	Design the χ^2 detector with consideration of sensor attacks and packet losses in CBTC systems.
[65], [66]	Statistical detection	KL divergence	Utilize the KL divergence as trustworthy factor in distributed power grid systems against corrupting sensor measurements.
[67]	Statistical detection	KL divergence	Detect and compensate thruster failures of autonomous underwater vehicle systems in real-time.
[68]	Learning-based detection	Supervised learning	Combine support vector machine technique and residual-based detector to detect sensor attack against building management systems.
[58]	Learning-based detection	Supervised learning	Generate anomaly detection model from input-output modeling of physical systems with the extreme learning machine model.
[69]	Learning-based detection	Supervised learning	Enhance the fault detection performance of χ^2 detector with convolutional neural network (CNN) model in automated vehicle systems.
[70]	Learning-based detection	Supervised learning	Adopt recurrent neural network (RNN) model for multiple sensor attack detection for vehicle systems.
[71]	Learning-based detection	Unsupervised learning	Adopt unsupervised dynamic Bayesian network model to detect cyber-physical attack in large-scale power grid system without data labeling.
[72]	Learning-based detection	Supervised & unsupervised learning	Utilize singular value decomposition to reduce computational costs, and detect power grid anomalies with SVM and clustering methods as supervised learning and unsupervised learning, respectively.

Algorithm 1 Residual-Based Anomaly Detection Algorithm

Input: $y(k)$
  $r(k) \leftarrow \|y(k) - C\hat{x}(k)\|$
  if $r(k) > \delta$ **then**
    $\Gamma(k) \leftarrow 1$
    Detect attack
  else
    $\Gamma(k) \leftarrow 0$
  end if
  Update time
  $\hat{x}(k+1) \leftarrow A_d\hat{x}(k) + B_d u(k) + L(y(k) - C_d\hat{x}(k))$
  $u(k+1) \leftarrow Py_r(k+1) - K\hat{x}(k+1)$

When a cyber-physical attack such as packet drop or packet manipulation is launched, there will be a mismatch between the sensor measurement $y(k)$ and the predicted sensor measurement $\hat{y}(k)$. However, anomalies such as packet transmission delays or sensor noise can also increase the residual $r(k)$ in (11). This may cause false positive detection of the attacks.

To overcome false detections, a residual-based detector sets up a detection threshold δ as follows:

$$\Gamma(k) = \begin{cases} 1, & \text{if } r(k) > \delta \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

When the residual $r(k)$ exceeds the threshold δ, the detection alarm $\Gamma(k)$ becomes true, and other resilient countermeasures handle the anomaly. We provide a pseudo code of the anomaly detection mechanism in Algorithm 1.

There is a trade-off between the anomaly detection speed and the detection accuracy with respect to the threshold level δ. A high-level threshold δ improves the detection accuracy by lowering the false-alarm rate from noise. However, it consumes more time from the moment the attack is launched until the residual $r(k)$ exceeds the threshold δ. In contrast, a low-level δ improves the detection speed but is vulnerable to false detection from noise. Therefore, it is necessary to select a proper threshold δ according to the attack types and system dynamics. The residual $r(k)$ rises in the detector from most cyber-physical attacks. However, for well-designed stealthy attacks, the residual $r(k)$ does not exceed the threshold δ, and consequently, the detector does not activate the attack alarm.

In [60], an empirical study that detects cyber-physical attacks with the residual-based anomaly detector is implemented in a water level-control testbed. In addition, the residual-based detector is adopted to reveal energy leakage due to a malicious user in a microgrid environment [73]. The residual-based anomaly detector with linear dynamics of UAV is proposed in [61], in which the authors adopt a real robotic vehicle to show various cyber-physical attacks, including sensor attacks, controller attacks, and control software manipulation. By considering the nonlinear dynamics of UAV, the extended Kalman filter (EKF)-based anomaly detector is proposed in [62], in which the authors compare sensor attack detection performance with the detector proposed in [61].

To enhance the security level against sophisticated cyber-physical attacks, a cumulative sum (CUSUM)-based anomaly detector is proposed in [74], which aggregates past residuals. The CUSUM statistic is recursively calculated with the

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

residual (11) as follows:

$$S(k) = \begin{cases} 0, & \text{if } k = 0 \\ max(0, S(k-1) + r(k) - \alpha), & \text{otherwise,} \end{cases}$$

where $S(k)$ is the CUSUM statistic, and α is a tuning parameter based on the noise of residual $r(k)$ during normal operation. By reflecting historical residuals, the CUSUM statistic can limit the impact of cyber-physical attacks better than the conventional residual (11).

From [23] and [63], a residual-based anomaly detector with the CUSUM reveals undetected attacks on a stateless residual-based detector (12), and the empirical studies in [23] and [63] validate CUSUM-based detection. In [75], an analysis of the worst-case attack impact on a CUSUM-based detector is provided by formulating a convex optimization problem, which shows the restriction on the attack impact on the CUSUM-based detector.

B. Statistical Detection

A χ^2 detector, known as the bad data detector, is one of the typical anomaly detectors adopted in various CPS, including power grid systems [55] and CBTC [64]. The χ^2 detector determines anomalies based on the statistical behavior of the physical system from sensor noises and network delays [76], [77]. Because most CPS operate in an environment in which sensor noise and network delays exist, the χ^2 detector is widely used to detect DoS attacks or bad-data injections [77], [78].

From the given system (7) and the observer (8), the χ^2 detector calculates the quantity $g(k)$ with the residue $z(k) = y(k) - C\hat{x}(k)$:

$$g(k) = z^T(k) Q^{-1} z(k), \tag{13}$$

where Q denotes the co-variance matrix of the residue $z(k)$. It is assumed that the residue $z(k)$ is zero-mean Gaussian and identically distributed for different k's [79].

When a random attack signal is injected by an attacker, the statistical property of the residue $z(k)$ changes, increasing the quantity $g(k)$. If the quantity $g(k)$ exceeds the threshold δ, the χ^2 detector determines there was an injection from a cyber-physical attack [80].

The χ^2 detector has a detection limit for historical data-based attack signals. When a sensor measurement from the past is injected into the anomaly detector, the χ^2 detector is unable to detect the attack because the noise of the recorded past sensor measurement was previously determined to be normal [19].

The χ^2 detector is adopted in a smart grid system [55], [56] and a CBTC system [64] to detect fault-data injection attacks damaging each system. In [55], a Euclidean distance based χ^2 detector is proposed to overcome the detection limit in a conventional χ^2 detector that cannot detect a sensor attack deceiving the state estimator. The cosine similarity-matching χ^2 detector proposed in [56] shows an advantage in the sensor attack detection time and enhanced detection performance against sophisticated sensor attacks. For sensor attacks on a CBTC system, the χ^2 detector proposed in [64] shows a higher success rate and a lower false-detection rate than the conventional intrusion detection system from packet monitoring.

The Kullback–Leibler (KL) divergence is deployed to reveal anomalies in sensor measurement, which compares probability distributions between benign and attacked sensor measurements [81], [82]. The KL divergence is denoted as follows:

$$D(\tilde{r}_k || r_k) = \int f_{\tilde{r}_k}(x) \log \frac{f_{\tilde{r}_k}(x)}{f_{r_k}(x)} dx, \tag{14}$$

where $\tilde{r}_k$ is a compromised residual, r_k is a residual (11) under normal operation, $f_{\tilde{r}_k}(x)$ and $f_{r_k}(x)$ are the probability density functions (PDFs) of the residuals $\tilde{r}_k$ and r_k, respectively. From the definition (14), the KL-divergence $D(\tilde{r}_k || r_k)$ has a non-negative value, in which the KL-divergence becomes zero only if $f_{\tilde{r}_k}(x) = f_{r_k}(x)$. In addition, although the KL divergence is a discrepancy metric between two PDFs, it is generally asymmetric, which means that $D(\tilde{r}_k || r_k) = D(r_k) || \tilde{r}_k)$ is not guaranteed.

When an arbitrary cyber-physical attack is launched, the PDF $f_{\tilde{r}_k}(x)$ under attack differs from the distribution of the PDF $f_{\tilde{r}_k}(x)$ of a benign residual r_k. If the KL-divergence $D(\tilde{r}_k || r_k)$ exceeds a certain value δ, it can be considered an attack situation [82]. The KL divergence-based sensor attack detection method is adopted in DC [65] and AC [66] microgrids, where the KL-divergence is utilized as a trustworthy factor of sensor measurements. A real-time fault detection method is proposed in [67], which considers the KL-divergence to determine the faults of an underwater autonomous vehicle control system.

C. Learning-Based Detection

Machine learning techniques are used for modeling complex CPS, supporting energy management, and controlling physical systems in the real world [83]. In the context of cyber-physical security, machine learning techniques can be utilized to distinguish between system faults and malicious attacks, to detect intrusions by unauthorized devices, and to design an optimal detector to minimize false detection. Data-driven cyber-physical security research has been conducted with enormous amounts of information from physical systems and networks in the various CPS.

Supervised learning methods utilize labeled data in normal and abnormal environments to train machine learning models; i.e., a detector using the supervised learning technique is learned based on the sensor measurements and control input signals that have been previously classified as normal and abnormal. In [68], incorporation of a model-based detector and a data-driven detector with supervised learning algorithms is proposed to detect stealthy sensor attacks targeting building energy-management systems, where offline learning of normal operation data was previously conducted for a support vector machine (SVM) model recognizing anomalies in the system.

A supervised learning-based anomaly detector with an extreme learning machine (ELM) model that learns control signals and sensor measurements of a power plant is proposed in [58]. The proposed detector is evaluated in a heavy-duty gas turbine hardware-in-the-loop testbed, which shows a low

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

TABLE V
LIST OF CYBER-PHYSICAL ATTACKS AND THEIR DETECTION METHODS

References	Attack type		Detection method	Key idea
[85]	DoS	Reactive jamming	Power envelop detection	Design an IEEE 802.15.4 header detector for the attacker to disturb only IEEE 802.15.4 packet transmissions.
[86]	DoS	Attack scheduling	Residual-based anomaly detector	Schedule jamming signal radiation sequences to maximize the impact of DoS attack with control-theoretical model
[87], [88]	DoS	Media vulnerability	Signal-to-noise ratio	Implement a remote jamming attack with jamming signal amplification by a signal repeater on waveguide media.
[89]	DoS	Protocol vulnerability	-	Deceive a transmitter by injecting well-crafted fake ACK packets to incapacitate the retransmission machanism of the IEEE 802.11 protocol.
[90]	DoS	-	CUSUM	Monitor vehicular traffic with the CUSUM algorithm on centralized road-side units in a vehicular ad hoc network (VANET).
[91]	DoS	-	Timestamp	Check if the packet transmission delay satisfies packet delivery deadline constraints with the time-stamp on the packet.
[92]	DoS	Reactive jamming	Signal strength	Propose an adaptive threshold to detect a jamming attack with the channel signal strength.
[93], [94]	Replay	-	Watermarking	Add an intentional Gaussian noise to control the input signal, and check the physical response for the injected noise.
[95]	ZDA	ZOH vulnerability	-	Generate a zero-dynamics attack targeting unstable sampling zeros created by periodical sampling in discrete-time control systems.
[52]	ZDA	Robust ZDA	-	Generate a ZDA signal with an imprecise physical system model, where the sniffed control input signal relaxes the system model knowledge constraint of the ZDA.
[96]	ZDA	-	Generalized hold	Replace ZOH with a generalized signal hold technique in discrete-time control systems.
[20], [97], [98]	ZDA		Coding	Modulate the control input signal or the sensor measurement to distort characteristics of ZDA
[99]	ZDA	ZOH vulnerability	Multirate sampling	Design a controller with multirate sampling to remove unstable sampling zeros.
[21]	PDA	Robust PDA	-	Generate a PDA signal with an imprecise physical system model, where the sniffed control input signal relaxes the system model knowledge constraint of the PDA.
[100]	PDA	-	Software-defined networking (SDN)	Sample the sensor measurement and detect the PDA on distributed network devices with control-theoretical calculation.
[101]	Covert	Data-driven method	-	Intercept both the control input signal and the sensor measurement, and compensate uncertainty of the physical system model of an attacker.
[53], [97]	Covert	-	Coding	Modulate the control input signal or the sensor measurement to distort characteristics of the ZDA

false-positive detection rate and a high true-positive detection rate. A convolutional neural network (CNN)-based sensor measurement screening method [69] is proposed to mitigate sensor faults in autonomous vehicles, in which the CNN model screens anomalous sensor readings, and the Kalman filter with the screened sensor measurement estimates the states of the vehicle. A recurrent neural network (RNN)-based sensor attack detection method for autonomous vehicles is proposed in [70] under conditions of uncertainty and nonlinearity, in which the proposed method identifies the combinations of attacked sensors when multiple sensors are attacked.

In contrast to supervised learning, unsupervised learning techniques do not require labeled data for training the model. Furthermore, unsupervised learning techniques categorize similar samples within the input data, a process known as clustering. Abnormal sensor measurements or control input signals are clustered differently from benign clusters. In addition, unsupervised learning techniques are also used in conjunction with supervised learning methods to improve the classification performance by extracting the internal features of sensor measurements and control input signals.

An unsupervised dynamic Bayesian network (DBN) model is proposed in [71] that detects abnormal sensor measurements for large-scale smart grids with time-varying energy information. For a sensor attack, the proposed DBN model achieves detection accuracy of 98%, and a false-positive rate of less than 2%.

An unsupervised machine learning-based fault detection method is proposed for heater, ventilation, and air conditioning (HVAC) systems [84]. In the proposed method, a Gaussian process estimates the state of the HVAC system, and an SVM model detects faults in the HVAC system. For sophisticated sensor-attack detection in distributed smart grid systems, machine-leaning-based detectors with supervised learning and unsupervised learning are proposed in [72], in which the authors use singular value decomposition of state observations to reduce the dimensions of the training data and the computational complexity.

VII. ANALYSIS OF CYBER-PHYSICAL ATTACKS

In this section, we analyze common cyber-physical attacks: the DoS attack, the replay attack, the zero-dynamics attack, the pole-dynamics attack, and the covert attack. We focus on the time response of physical systems and the stealthiness of cyber-physical attacks. For a clear understanding of the impact of each attack, we provide detailed numerical examples for each attack. Furthermore, we provide an analysis of advanced cyber-physical attacks and detection methods for each attack. Table V summarizes improvements and detection methods for cyber-physical attacks in Table III.

A. Denial-of-Service (DoS) Attack

A DoS attack is one of the typical network attacks that disturb packet transmissions by exploiting network

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

vulnerabilities. A DoS attack mainly causes two communication faults: packet drops and network transmission delays. Packet dropping is one of the most common DoS attack methods. Over the past decades, numerous DoS attack studies have been implemented in each layer of the OSI network model. Hence, it is formidable to cover overall DoS attack mechanisms in a subsection. Comprehensive surveys of DoS attacks can be found in [17], [102]. In this survey, we briefly introduce common DoS methodologies in conventional networks.

In the physical layer of the OSI network model, packet dropping is implemented by link cut-off and the emission of high-gain jamming signals against wired and wireless networks [103]. In logical ways, it is implemented by MITM attacks and well-crafted packet injections with protocol vulnerabilities. Address resolution protocol (ARP) and Internet protocol (IP) can be spoofing targets in the data link layer and the network layer of the OSI network model, respectively [17]. ARP and IP spoofing are the typical spoofing attacks that steal packets on the network by deceiving communication nodes. Also, the spoofing attacker can destroy some packet fields related to packet error detection, such as checksum, resulting in invalidation of received packets.

Furthermore, a DoS attack can incur significant delays by exhausting the network resources. DoS attacks such as ICMP flooding [38], UDP flooding, and TCP connection flooding exhaust network resources by maliciously enqueuing the memory in network equipment or requesting excessive responses from the computing system, which result in transmission delays and time-outs. In particular, most DoS attacks in the application layer of the OSI network model target protocol vulnerabilities, resulting in exhaustion of network resources and computing resources. The attacker can incur massive packet generation by exploiting a vulnerability in the domain name system (DNS) protocol related to the recursive query property. Furthermore, an attacker can exhaust the computing resources of the computing system by generating deformed packets to exploit vulnerabilities in hypertext protocol (HTTP) [102].

A DoS attack from the CPS perspective is the simplest to implement among the attacks shown in Table III because an accurate mathematical model of the physical system is not required [12], [104]. The physical system can become unstable by DoS attacks, which cause packet drops and network delays. Many control theoretical studies analyze a relationship between the stability of the physical system and packet drops in the network [86]. Furthermore, the delay bounds for physical stability and delay-induced control performance degradation are theoretically quantified in [105]. However, in practical CPS, an excessive network delay is considered a transmission time-out, and the physical systems hold the previous control input signal or switch to the fail-safe mode. Consequently, excessive network delays caused by DoS attacks are considered packet drops in the network.

In NCS, packet drops due to a DoS attack may cause degradation in control performance and system instability [106]. A DoS attack can be described by a packet drop model [86], where transmission failure is generated by packet drops only on the synchronized NCS in the discrete-time domain, as shown in Fig. 8. If a packet drop occurs as shown in Fig. 8 at time k, it means that neither the sensor measurement $y(k)$ nor the control input signal $u(k)$ is transmitted. The packet drop is expressed by the following packet drop model:

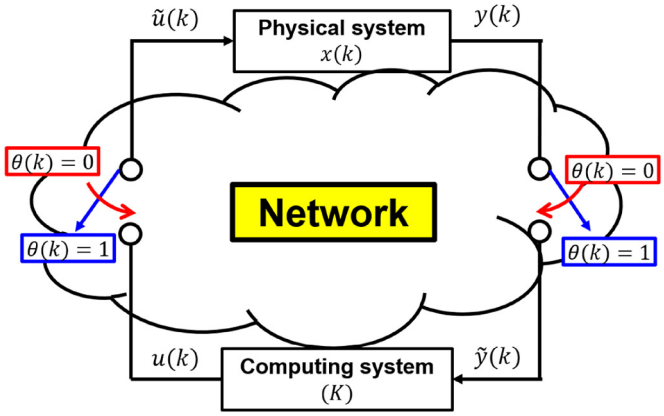


Fig. 8. Switched network DoS attack model.

$$\theta(k) = \begin{cases} 1, & \text{with packet drop} \\ 0, & \text{without packet drop,} \end{cases} \tag{15}$$

where $\theta(k)$ is the packet drop indicator. The packet drop indicator $\theta(k) = 1$ represents packet transmission failure in the network from a DoS attack.

When packet transmission fails due to a DoS attack, the physical system and the computing system cannot receive control input signals and sensor measurements, respectively. According to the updated failure of a control input signal from a DoS attack, the physical system is actuated from the past control-input signal with the ZOH as follows:

$$x(k+1) = \begin{cases} A_d x(k) + B_d u(k), & \text{if } \theta(k) = 0 \\ A_d x(k) + B_d u(k-1), & \text{if } \theta(k) = 1. \end{cases} \tag{16}$$

In addition, the state estimation of the computing system depends on the physical system model by considering the packet drop as follows:

$$\hat{x}(k+1) = \begin{cases} A_d \hat{x}(k) + B_d u(k) \\ +L(y(k) - C\hat{x}(k)), & \text{if } \theta(k) = 0 \\ A_d \hat{x}(k) + B_d u(k-1), & \text{if } \theta(k) = 1. \end{cases} \tag{17}$$

If there is no transmission delay or sensor uncertainty, regardless of a DoS attack, state estimation (17) completely traces the state of the physical system $x(k)$. However, due to the integration of noise in NCS by a persistent DoS attack, the state estimation error between the physical state $x(k)$ and the estimation $\hat{x}(k)$ is increased, which causes a deterioration of control performance [86].

DoS attacks are characterized by the duration of the attack and the frequency with which it occurs within a specific time interval [107]. Fig. 9 shows DoS attacks in the time interval $[i, j]$. The duration of an attack is defined as the time duration $[a_k, a_k + \tau_k]$, where a_k is the start time of the k-th DoS attack and τ_k is the duration of the k-th DoS attack.

1) Attack Constraints: To implement a DoS attack, it is necessary for the attacker to access the communication channel between the physical and computing systems. In addition, the

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

Fig. 9. The duration of a DoS attack.

attacker intentionally incurs packet exchange with proper DoS attack methods including jamming signals, flooding, or well-crafted packets exploiting network vulnerabilities. However, in terms of actual attack implementation, the attacker also has energy constraints based on hardware performance. In [18], a hardware limitation of the attacker is modeled as the scheduling of packet drops in a certain time interval, illustrated in Fig. 9. The attacker determines time slots to drop the packets with the following two constraints: the number of packet drop attempts and the number of consecutive packet drop time slots in a packet drop attempt. In other words, packets cannot always be dropped if the attacker has a hardware constraint.

2) Numerical Example: We consider the physical system (6) as follows:

$$A \triangleq \begin{bmatrix} 0 & 1 \\ -2 & 1 \end{bmatrix}, \quad B \triangleq \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C \triangleq \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{18}$$

From the system dynamics (18), the transfer function form of the system (6) is presented as follows:

$$G(s) = \frac{s-1}{s^2 - s + 2}. \tag{19}$$

The transfer function (19) shows that the physical system (6) has two unstable poles, $p = 0.5 \pm j1.32$, and one unstable zero, $z = 1$, on the right-half plane. Furthermore, we properly design the controller gain K and the observer gain L to bring the state $x(t)$ to zero.

We implement the MATLAB simulation shown in Fig. 10 to illustrate the impact of the attack on the physical system (6) with the dynamics (18) when a DoS attack is launched on the network in the continuous-time domain. In addition, we compare the control input signal $u(t)$ and the sensor measurement $y(t)$ in an attack-free environment to a DoS attack.

When we launch the DoS attack on the network at time $t = 1.5$ s, the physical state $x(t)$ diverges to infinity by the physical impact (16). In other words, the state response to the DoS attack is equal to the open-loop system without any controller.

The control input signal $u(t)$ cannot be transmitted to the physical system owing to the DoS attack. Therefore, the control input signal applied to the physical system is held by the ZOH from the start of the DoS attack. As illustrated in the second graph of Fig. 10, the control input signal $u(t)$ on the physical system does not converge to the control input $u_{ref}(t)$ in an attack-free environment.

Similarly, as the control input signal, the sensor measurement $y(t)$ does not enter the computing system under the DoS attack. From packet drops in the network, the sensor measurement $y(t)$ on the computing system is held by the ZOH, as illustrated in the third graph of Fig. 10, which does not converge to the sensor measurement $y_{ref}(t)$ in an attack-free environment.

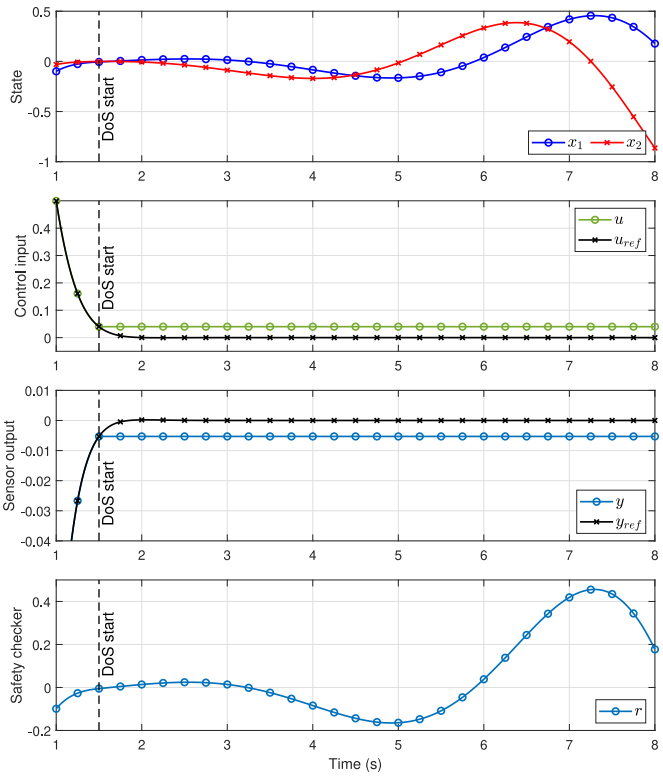


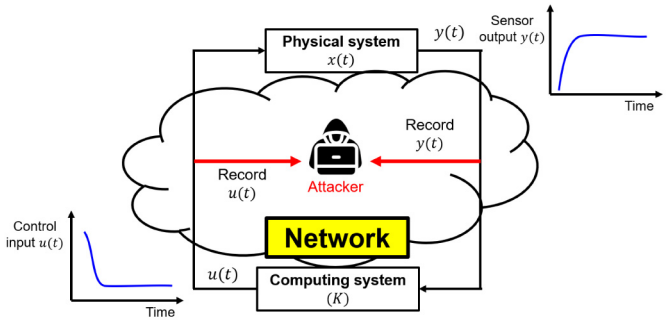
Fig. 10. Numerical example of the DoS attack.

We assume that the anomaly detector recognizes the DoS attack at the start of the attack because the sensor measurement packets are not continuously transmitted to the computing system. Then, the safety checker in Fig. 7 estimates the expected sensor measurement $\hat{y}(t)$, where $\hat{y}(t)$ diverges to infinity. Therefore, the safety checker determines that the physical state is going to violate the allowable region. The simulation result shows that a sufficiently long attack duration is required for the physical state $x(t)$ to violate the allowable region, as illustrated in Fig. 10. However, because the DoS attack is detected by the safety checker, the attack can be handled with certain countermeasures.

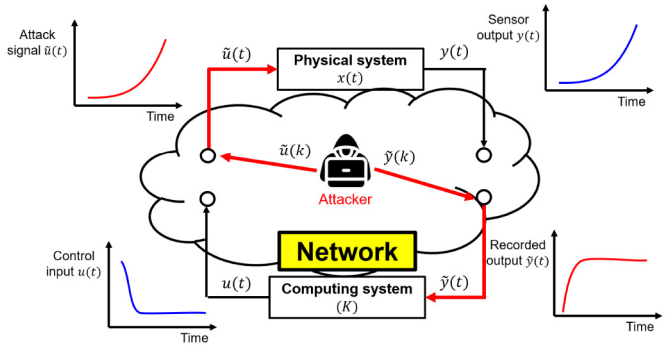
3) Improved Attacks: Reactive jamming [85] is one of the optimal jamming attacks, in which the attacker listens to the communication channel and reacts by colliding with the transmitted signal. The reactive jamming attack has advantages for hard detection and energy-efficient communication disturbances. An optimal DoS attack scheduling [86] is proposed from the viewpoint of the attacker with a power constraint, which shows that consecutive DoS attacks can inflict more damage on the physical system.

A jamming attack targeting a communication based train control (CBTC) system is proposed in [87], [88], where a jamming attack is amplified by a repeater on the CBTC system. As a result, the intermittent communication disruption caused by the attack degrades train operations. An intelligent jamming attack targeting IEEE 802.11 wireless networks is proposed in [89], in which the transmitter is deceived with a fake ACK frame while causing packet collisions with jamming signals.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.



(a) Recording phase



(b) Injecting phase

Fig. 11. Illustration of the replay attack.

4) Detection Strategy: For a vehicular ad hoc network (VANET), a real-time DoS attack-detection framework is proposed in [90], in which a road-side unit (RSU) monitors the vehicle traffic with a CUSUM and the geometric entropy-minimization algorithm to detect the DoS attack. A timestamp-based jamming attack detection scheme is proposed in [91], in which communication disturbances are detected when the timestamp of the packet exceeds the delivery deadline in clustered sensor networks. To detect reactive jamming, the authors in [92] propose a reactive jamming attack-detection algorithm with an adaptive threshold while periodically checking the signal strength and packet consistency.

Various DoS attacks, such as powerful jamming signal radiation in physical communication, traffic flooding in each layer of the OSI network model, and exploiting application vulnerabilities, can occur in the CPS networks. Regardless of DoS attack types, the conventional anomaly detection methods, illustrated in Section VI, can only detect whether the DoS attack is forced or not. Therefore, it is hard for physics-based anomaly detection methods to specify the DoS attack type. To handle the DoS attack after the detection, DoS attack analysis in the OSI network model should be accompanied by a physics-based anomaly detector.

B. Replay Attack

The replay attack [19] is a stealthy cyber-physical attack in which an intruder injects past sensor measurements into a computing system, as shown in Fig. 11. To implement the attack, the intruder must be able to break into both the sensor measurement and control input signal transmission channels. In contrast to other stealthy attacks, the replay attack does not require the physical system model. Instead, the replay attack consists of a recording phase and an injecting phase.

In the recording phase, the attacker only breaks into the sensor measurement transmission channel. Then, the attacker sniffs and records the sensor measurement $y(k)$ as illustrated in Fig. 11(a). Because there is no manipulation of the sensor measurement and the control input signal in this phase, the physical systems operate normally.

In the injecting phase, however, the attacker hijacks both the sensor measurement and control signal transmission channels simultaneously, as illustrated in Fig. 11(b). Then, the attacker injects the previously recorded sensor measurement $\tilde{y}(k)$ into the computing system. At the same time, the attacker injects the malicious control input signal $\tilde{u}(k)$, which renders the physical system unstable.

The anomaly detector determines that the recorded sensor measurement $\tilde{y}(k)$ is intact because the past sensor measurement equals the time-shifted sensor measurement from the physical system. Therefore, the sensor measurement of the attacker, $\tilde{y}(k)$, does not violate the model-based anomaly detector. In addition, the χ^2 detector cannot detect a replay attack because the noise of the past sensor measurement $\tilde{y}(k)$ is already accepted. When the deviation between the original sensor measurement $y(k)$ and the recorded sensor measurement $\tilde{y}(k)$ is small enough at the start of the attack, the residual, $r(k)$, converges to zero. Therefore, the replay attack has the property of stealthiness.

1) Attack Constraints: To implement the recording phase and the injecting phase, it is necessary for the attacker to hijack both transmission channels for the sensor measurement $y(k)$ and the control input signal $u(k)$. In the recording phase, the attacker must have enough memory to record the sensor measurement $y(k)$, which is related to the attack duration; a successive replay attack requires enough recording time and sensor samples to inject successive past sensor measurements. During the injection phase, the attacker's control input signal, $\tilde{u}(k)$, is designed to cause the state of the physical system to diverge.

2) Numerical Example: We implement a simulation study of the replay attack against a physical system with the dynamics (18), as shown in Fig. 12. In this simulation, the recording phase is the time interval $t = [2, 5]$ s, and the injecting phase is the time interval $t = [5, 8]$ s.

At the start of the injecting phase, $t = 5$ s, the state of the physical system, $x(t)$, becomes unstable because of the false control input signal $\tilde{u}(t)$. However, the state $x(t)$ converges to zero during the recording phase.

Following the injection of a malicious control input $\tilde{u}(t) = 0.001$, the control input signal $u(t)$ of the controller experiences a temporary perturbation because of the difference between the original sensor measurement $y(t)$ and the recorded sensor measurement $\tilde{y}(t)$. However, the control input signal $u(t)$ of the controller converges to the control input signal in the attack-free environment, $u_{ref}(t)$.

The recorded sensor measurement, $\tilde{y}(t)$, from the recording phase $t = [2, 5]$ s is injected into the computing system during the injecting phase $t = [5, 8]$ s, as shown in Fig. 12, where

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

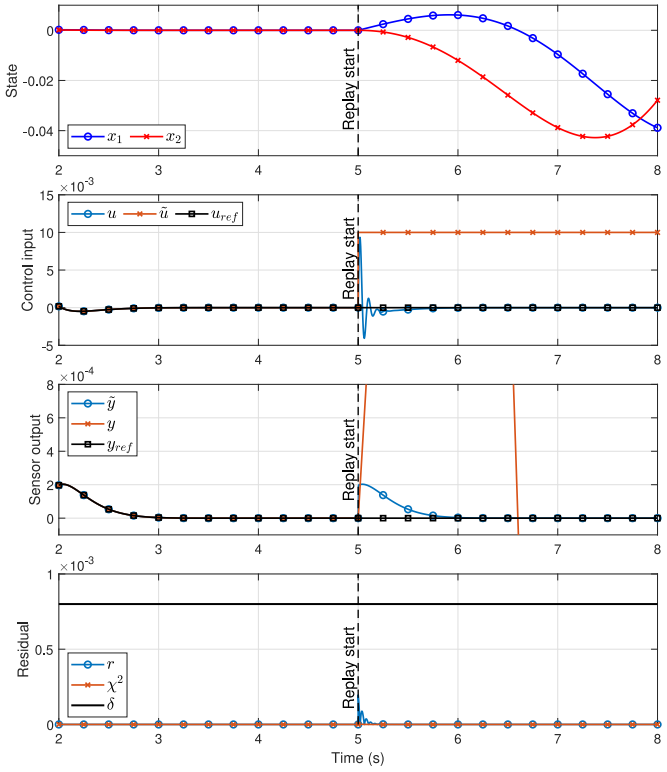


Fig. 12. Numerical example of the replay attack.

the sensor measurement $\tilde{y}(t)$ traces the sensor measurement $y_{ref}(t)$ in the attack-free environment. However, the real sensor measurement, $y(t)$, deviates because a malicious control input signal $\tilde{u}(t)$ is injected into the physical system.

We evaluate the stealthiness of the replay attack with the residual-based detector and the χ^2 detector. The residual $r(t)$ in the fourth graph of Fig. 12 shows the stealthiness of the replay attack. In Fig. 12, we set the threshold $\delta = 0.0008$ to detect a replay attack. The residual $r(t)$ slightly rises at time $t = 5$ s owing to the deviation between the original sensor measurement $y(t)$ and the recorded sensor measurement $\tilde{y}(k)$. However, the residual-based detector perceives the replay attack injection $\tilde{y}(t)$ as an attack-free environment because of the residual vanishing. Furthermore, the quantity $g(k)$ of the χ^2 detector (13) has no perturbation against the deviation by the replay attack, as shown in the fourth graph of Fig. 12. Therefore, the attack alarm is not activated. Consequently, the simulation result in Fig. 12 shows the stealthiness of the replay attack.

3) Improved Attacks: To the best of our knowledge, no studies have included advanced replay attacks within their scope.

4) Detection Strategy: A physical watermarking method to detect the replay attack is proposed in [93]. The controller in the computing system adds a Gaussian watermarking signal $\zeta(k)$ to the control input signal $u(k)$ and uses the Kalman filter to predict the state of the physical system in the next time step. Then, the physical system returns the sensor measurement $y(k)$, including the response to the watermarking

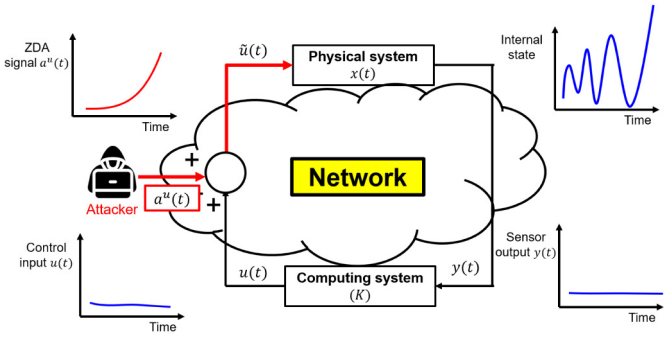


Fig. 13. Illustration of the zero-dynamics attack.

signal $\zeta(k)$. As a result, the watermarking signal is returned to the computing system. When the replay attack is launched, the previously recorded sensor measurement $\tilde{y}(k)$ is injected into the computing system. However, the previously recorded sensor measurement $\tilde{y}(k)$ includes the physical response for the old watermarking signal, which differs from the current watermarking signal $\zeta(t)$. Therefore, the residual $r(k)$ increases because of the unmatched watermarking signal $\zeta(t)$, and finally, the anomaly detector detects the replay attack.

But, the watermarking method has downsides. The watermarking signal can degrade the control performance of the physical system and can activate a false-positive detection alarm. In [94], an optimal watermarking signal scheduling strategy is proposed to reduce the false-positive alarm rate and the deterioration of control performance.

C. Zero-Dynamics Attack

The zero-dynamics attack (ZDA) [12], [20] is a control input injection attack that exploits the zero dynamics of a system with a non-minimum phase zero. To realize a zero-dynamics attack, the attacker needs to know the zero dynamics of the target system and the specific control input that makes the output $y(t) = 0$. Then, the attacker can render a certain internal state of the system diverge by adding the specific input to the ordinary input signal, as illustrated in Fig. 13.

It should be noted that the ZDA cannot be detected by observing the output $y(t)$ because the effect of the ZDA on the output is zero. One may think that the ZDA is limited only to the system with an intrinsic non-minimum phase zero. However, discretization of a continuous system for digital control can introduce an additional non-minimum phase zero to the system, which can be exploited by the attacker [95].

The ZDA is a stealthy controller attack that targets unstable zeros on the right-half plane in the s-domain. The ZDA is represented as a controller attack because it adds an attack signal $a^u(t)$ to the control input signal $u(t)$ in the physical system (6) as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B(u(t) + a^u(t)) \\ y(t) &= Cx(t). \end{aligned} \tag{20}$$

To consider the zero-dynamics of the physical system, we rewrite the system (20) into the Byrnes–Isidori normal form

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

as follows:

$$\begin{aligned}\begin{bmatrix}\dot{z_s}(t)\\ \dot{z_u}(t)\end{bmatrix} &= \begin{bmatrix}S_s & 0\\ 0 & S_u\end{bmatrix}\begin{bmatrix}z_s(t)\\ z_u(t)\end{bmatrix} + \begin{bmatrix}P_s\\ P_u\end{bmatrix} y(t)\\ \dot{\eta}(t) &= A_v\eta(t) + B_v(\psi^T\eta(t) + \phi_s^T z_s(t) + \phi_u^T z_u(t)\\ &\quad + g(u(t) + a^u(t)))\\ y(t) &= C_v\eta(t), \end{aligned} \tag{21}$$

where S_s, S_u, P_s, P_u, ψ, ϕ are constant matrices with suitable dimensions, and the eigenvalues of matrices S_s and S_u are on the left-half plane and the right-half plane, respectively. For constant $i \geq 1$, matrices A_v, B_v, and Cv can be described as follows:

$$A_v = \begin{bmatrix}0_{i-1} & I_{i-1}\\ 0 & 0_{i-1}^T\end{bmatrix}, \quad B_v = \begin{bmatrix}0_{i-1}\\ 1\end{bmatrix}, \quad C_v = \begin{bmatrix}1 & 0_{i-1}^T\end{bmatrix}.$$

From the given system (21), we can design the ZDA as follows:

$$\dot{z^a}(t) = S_u z^a(t), \quad a^u(t) = -\frac{1}{g}\phi_u^T z^a(t), \tag{22}$$

where $z^a(t)$ is the zero-dynamics of the attacker. Owing to the eigenvalue of matrix S_u in the right-half plane, the zero-dynamics $z^a(t)$ diverge to infinity, and thus the attack signal $a^u(t)$ from (22) diverges. From the combination of the system (21) and the ZDA (22), we confirm that the ZDA signal $a^u(t)$ negates the unstable zero-dynamics term $\phi_u^T z_u(t)$, where the attacked η dynamics and the sensor measurement $y(t)$ remain stable. However, in terms of the system (20), the state $x(t)$ diverges to infinity because of the divergence of the ZDA signal $a^u(t)$.

Although the ZDA signal diverges to infinity, the stable form of the sensor measurement $y(t)$ from unstable zeros causes an estimation fault in the state estimator, i.e., the state estimator calculates the state estimation $\hat{x}(t)$ as stable though the real physical state, $x(t)$, becomes unstable. From the estimation error for $\hat{x}(t)$, the anomaly detector determines that the sensor measurement $y(t)$ is normal in spite of the ZDA and does not activate the attack alarm. Therefore, the ZDA is stealthy.

1) Attack Constraints: To implement the ZDA, the attacker must hijack the control input transmission channel on the network because the ZDA is a controller attack that manipulates the control input signal $u(t)$.

For the ZDA to destabilize the physical state $x(t)$, the target physical system must have at least one zero in the right-half plane. If all the zeros of the physical system are in the left-half plane, the impact of the ZDA is eliminated over time. To design the ZDA, it is necessary for the attacker to have an accurate model of the physical system (6). When the attacker has an inaccurate model of the target physical system, the attacker can only launch a poorly designed ZDA that cannot fully eliminate the $\phi_u^T z_u(t)$ term in (21). Then, the ZDA will be detected by the anomaly detector.

To allow the ZDA to render the state $x(t)$ diverge while maintaining stealthiness, the actuator of the physical system must not be saturated by the ZDA signal $a^u(t)$. Because every actuator driving the real physical system has a control input limit, a ZDA signal greater than the input limit will saturate the control input of the physical system. Then, the elimination of the $\phi_u^T z_u(t)$ term by the ZDA will fail. Consequently, the ZDA is detected by the anomaly detector when the control input is saturated.

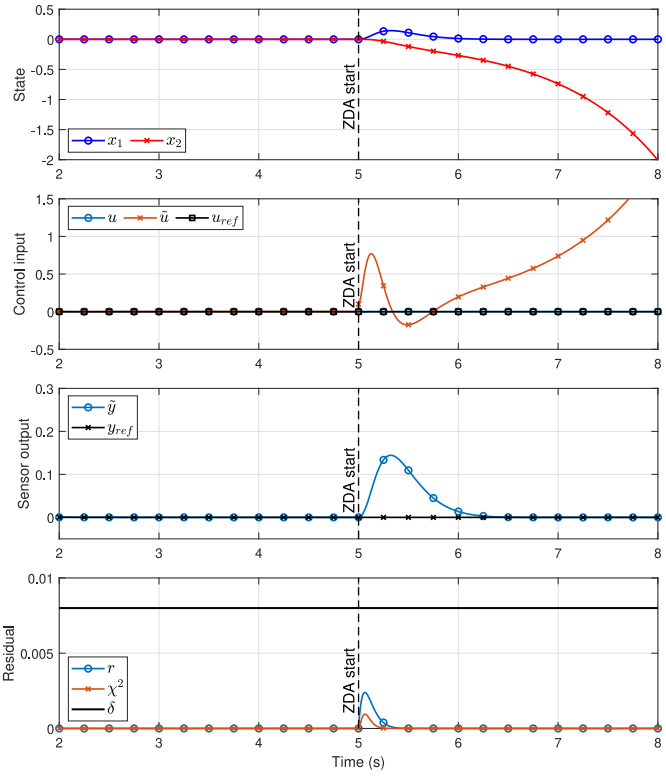


Fig. 14. Numerical example of the zero-dynamics attack.

2) Numerical Example: We implement a simulation study of the ZDA launched on a physical system with the dynamics (18) as shown in Fig. 14, where the ZDA is injected at time $t = 5$ s. Because of the zero dynamics, the state $x_2(t)$ diverges whereas the state $x_1(t)$ converges to zero with a short-term perturbation after the ZDA launches.

The ZDA signal $a^u(t)$ is designed with an intrinsic zero $z = 1$ in the right-half plane. The attacker starts to add $a^u(t)$ to the control input signal $u(t)$ from the computing system at time $t = 5$ s. We confirm that the modified control input $\tilde{u}(t)$ diverges in contrast to the converging control input signal $u_{ref}(t)$ in an attack-free environment. The received sensor measurement, $y(t)$, also has a small perturbation at the start of the ZDA, but the sensor measurement $y(t)$ converges to $y_{ref}(t)$ in an attack-free environment over time.

We evaluate the stealthiness of the ZDA with the residual-based detector and the χ^2 detector. The residual $r(t)$ in the fourth graph of Fig. 14 shows the stealthiness of the ZDA, where we set the threshold $\delta = 0.008$ to detect the attack. The residual $r(t)$ also converges to zero with some perturbation at the start of the ZDA. This small residual $r(t)$ below the threshold δ allows the ZDA to be stealthy. Similarly, the quantity $g(t)$ in the χ^2 detector (13) shows small perturbation at the start of the attack. However, the $g(t)$ does not exceed the threshold δ.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

3) Improved Attacks: Because of the conversion from a continuous physical system (6) to a discrete physical model (7) with periodical sensor measurement sampling, additional unstable sampling zeros can be added to the system model (7) with the ZOH [99]. In [95], the authors propose a ZDA targeting the sampling zeros. It is shown that the attacker can launch a ZDA for a physical system that does not have intrinsic zeros if the attacker knows the system dynamics (6) and the sampling rate of CPS.

Implementation of a ZDA requires the attacker to have the precise dynamics of the physical system. Therefore, it is difficult to implement a ZDA in practice. However, a robust ZDA proposed in [52] enables the attacker to design a ZDA signal with uncertain system dynamics if the attacker can eavesdrop on the sensor output $y(t)$ on the network. A disturbance observer of the attacker compensates for uncertainty about system dynamics with disclosure resources, which provides stealthiness for the robust ZDA.

4) Detection Strategy: A generalized hold method that replaces the ZOH is proposed in [96]. A generalized hold in a discrete-time control system can place the zeros at any desired positions. Therefore, when the attacker launches a ZDA targeting the unstable zeros, assuming a conventional ZOH system, the targeted zero of the ZDA signal does not match the zero replaced by the generalized hold. Therefore, the generalized hold method reveals the ZDA in a discrete-time control system.

System modulation strategies are proposed to detect stealthy cyber-physical attacks, including the ZDA. Modulations of the system matrix are proposed in [20] and [97], where the system dynamics altered by a modulation matrix distorts the ZDA signal of the attacker. Consequently, the ZDA is detected by the anomaly detector due to an increased residual $r(t)$ above the threshold. In [98], a two-way coding method that transforms the control input signal $u(t)$ and the sensor measurement $y(t)$ is proposed as a ZDA detection strategy. When the ZDA signal $u^a(t)$ is injected into a network equipped with the proposed two-way coding, the ZDA signal is distorted. Because of the distortion of the ZDA, the computing system can detect the ZDA, and correct the manipulated control input signal $\tilde{u}(t)$.

In [99], a multi-rate adaptive controller is adopted to detect a ZDA for discrete-time feedback control systems. The computing system with a state estimation rate that is faster than the control input generation rate will detect the anomaly from the sensor measurement $y(t)$. The experimental results in [99] show that the proposed multi-rate adaptive controller recovers a quadroter that is under a ZDA and triggers safe model control.

D. Pole-Dynamics Attack

The pole-dynamics attack (PDA) [21] is a sensor measurement modification attack to deceive the controller by eliminating the divergence of the unstable poles in the output sensor measurement, as illustrated in Fig. 15. Through the PDA, the controller misconceives that the physical system remains stable, although the physical system is actually becoming unstable. The PDA can be considered as a dual of the ZDA in that the PDA exploits the unstable pole dynamics by deceiving the sensors, whereas the ZDA exploits the unstable zero dynamics by deceiving the actuators.

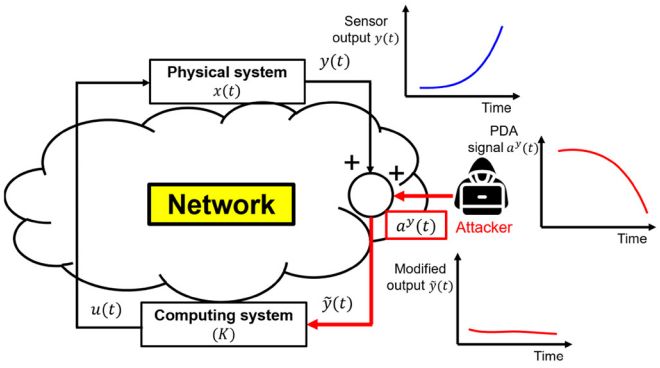


Fig. 15. Illustration of the pole-dynamics attack.

The PDA is a stealthy sensor attack that targets the pole dynamics in the right-half plane. We rewrite the physical system (6) adding the PDA signal as follows:

$$\begin{bmatrix}\dot{x_s}(t)\\ \dot{x_u}(t)\end{bmatrix} = \begin{bmatrix}A_s & 0\\ 0 & A_u\end{bmatrix}\begin{bmatrix}x_s(t)\\ x_u(t)\end{bmatrix} + \begin{bmatrix}B_s\\ B_u\end{bmatrix}u(t)$$
$$y(t) = C_s x_s(t) + C_u x_u(t) + a^y(t), \tag{23}$$

where matrices A_s, A_u, B_s, B_u, C_s, and C_u have proper dimensions, and the eigenvalues of matrices A_s and A_u are in the left-half plane and the right-half plane, respectively. We assume that the system (23) has at least one unstable pole, which is described by A_u.

From the given system (23), the PDA signal is designed as follows:

$$\dot{x}^a(t) = A_u x^a(t), \quad a^y(t) = -C_u x^a(t), \tag{24}$$

where $x^a(t)$ is the state of the attacker. Because of the eigenvalues of the matrix A_u in the right-half plane, the PDA signal $a^y(t)$ diverges to infinity. From the combination of the system (23) and the PDA (24), we can confirm that the PDA signal $a^y(t)$ eliminates the unstable pole-dynamics term $C_u x_u(t)$ in the sensor measurement $y(t)$. In [21], elimination of the pole-dynamics is represented as follows:

$$\lim_{t\to\infty}[x(t) - x^a(t)] = 0, \tag{25}$$

which means that the physical state $x(t)$ diverges to infinity because $x^a(t)$ diverges to infinity via the matrix A_u.

In the computing system, the state estimator incorrectly estimates the state $x(t)$ as $x(t) - x^a(t)$ based on the manipulated sensor measurement $\tilde{y}(t)$. Thus, the expected sensor measurement $C\hat{x}(t)$ of the state estimator traces the manipulated sensor measurement $\tilde{y}(t)$.

As a result of incorrect tracing of the expected sensor measurement $C\hat{x}(t)$ owing to the PDA, the residual $r(t)$ converges to zero over time. Therefore, the anomaly detector does not activate the attack alarm despite the PDA. Consequently, the PDA is stealthy with an accurate system model.

1) Attack Constraints: To implement the PDA, the attacker must hijack the sensor measurement transmission channel on the network because the PDA is a sensor attack that manipulates sensor measurement $y(t)$. For the PDA to destabilize

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

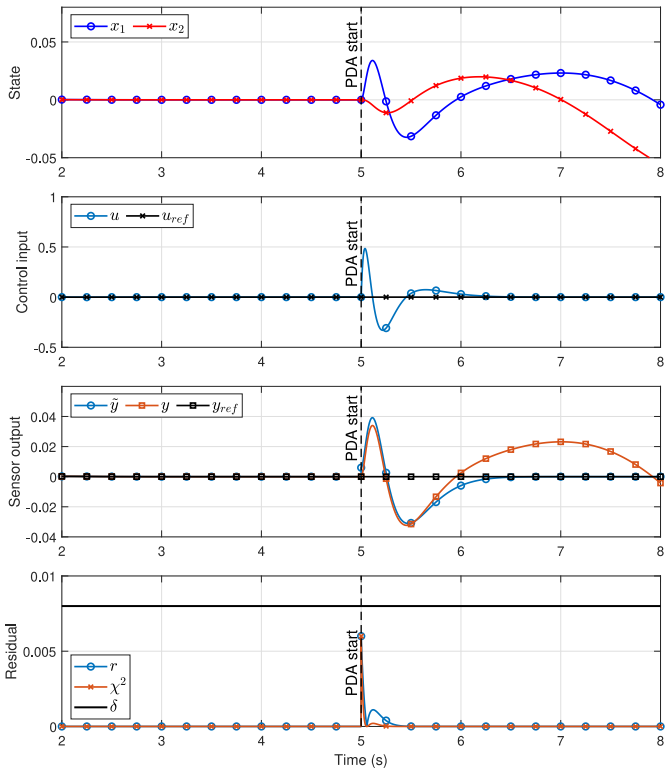


Fig. 16. Numerical example of the pole-dynamics attack.

the physical state $x(t)$, the target physical system must have at least one pole in the right-half plane. If all the poles of the physical system are in the left-half plane, the impact of the PDA is diminished over time. In addition, it is necessary for the attacker to know the system matrix A and the output matrix C of the target physical system. If the PDA is designed with inaccurate system poles, the condition on the elimination for the pole-dynamics (25) cannot be satisfied. This discrepancy will cause the residual $r(t)$ to rise, and the PDA will be detected by the anomaly detector.

2) Numerical Example: We implement a simulation study of the PDA launched for a physical system with the dynamics (18) as shown in Fig. 16, where (18) has two inherently unstable poles of $p = 0.5 \pm j1.32$ as shown in (19). We inject the PDA at time $t = 5$ s. From the start of the PDA at $t = 5$ s, the state of the physical system, $x(t)$, diverges, as illustrated in Fig. 16. The state transition by the PDA shows that the PDA has disruption resources that can damage the physical system.

Due to the distinction between the state of the physical system, $x(t)$ at $t = 5$ s, and the initial state of the attacker, $x^a(t)$, there is a temporary perturbation in the control input signal $u(t)$. However, the control input signal $u(t)$ asymptotically converges to the control input signal $u_{ref}(t)$ in an attack-free environment over time.

The original sensor measurement $y(t)$ from the physical system diverges to infinity via the PDA. However, the elimination of pole-dynamics in (25) confirms that the manipulated sensor measurement $\tilde{y}(t)$ converges to the sensor measurement $y_{ref}(t)$ in an attack-free environment, as shown in Fig. 16.

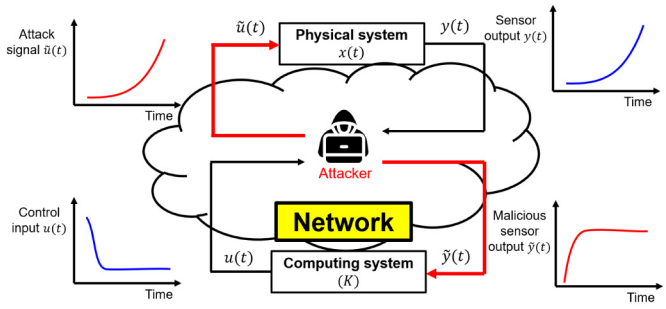


Fig. 17. Illustration of the covert attack.

We evaluate the stealthiness of the PDA with the residual-based detector and the χ^2 detector. The residual $r(t)$ in the fourth graph of Fig. 16 shows the stealthiness of the PDA. Here, we set the threshold $\delta = 0.008$ to detect the attack. Because of the state of the physical system, $x(t)$, and the state of the attacker, $x^a(t)$, at the start of the attack, $t = 5$ s, the temporary perturbation is generated on the residual $r(t)$. However, because of the estimation error by the state estimator, the residual $r(t)$ does not exceed the threshold δ, and thus no attack alarm is generated by the residual-based detector. Similarly, the χ^2 detector cannot generate an attack alarm because the quantity $g(t)$ in (13) does not exceed the threshold δ, as shown in the fourth graph of Fig. 16. Therefore, the PDA is also stealthy with respect to the χ^2 detector.

3) Improved Attacks: The design of the PDA signal $y^a(k)$ requires a precise model of the physical system, rendering it difficult to launch the PDA on the network. However, the robust PDA proposed in [21] enables an attacker with uncertain system dynamics to inject a stealthy attack and maintain stealthiness. Similarly to the robust ZDA [52], an attacker with disclosure resources sniffs the control input signal $u(t)$ on the network. The disturbance observer of the attacker compensates for the mismatch between the actual physical dynamics and the nominal dynamics of the attacker by utilizing the sniffed control input signal $u(t)$.

4) Detection Strategy: A resilient CPS architecture with a dual anomaly detector algorithm [100] has recently been proposed for real-time detection of the PDA launched through the network, where the dual anomaly detector algorithm is embedded on the distributed network devices. In the proposed architecture, the network devices periodically collect and inspect the sensor measurements with the embedded detection algorithm. When the PDA is detected, the network devices notify the centralized network manager of the PDA, and the network manager isolates the network attack point of the PDA and reconstructs the network paths for recovering the control performance of CPS.

E. Covert Attack

The covert attack is a stealthy, combined sensor and controller attack that takes over control authority of the physical system and deceives the computing system, simultaneously, as illustrated in Fig. 17. In contrast to the replay attack, which only sniffs the sensor measurement $y(t)$, the covert attack requires sniffing both the control input signal $u(t)$ and

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

the sensor measurement $y(t)$. The covert attack consists of two processes (deception and injection). Under the covert attack, these two processes are executed at the same time, i.e., deceiving the computing system by manipulating sensor measurements and injecting malicious control inputs into the physical system.

In the deception process, the attacker hijacks the sensor measurement transmission channel to inject the malicious sensor measurement $\tilde{y}(t)$, which deceives the computing system. To design the covert sensor attack signal, we assume that the attacker has a virtual physical system with a precise model of the physical system. From the control input signal's transmission channel, the attacker sniffs the control input signal $u(t)$ on the network and calculates the state of the attacker, $x^a(t)$, in the virtual physical system as follows:

$$\begin{aligned} \dot{x}^a(t) &= Ax^a(t) + Bu(t) \\ \tilde{y}(t) &= Cx^a(t). \end{aligned} \tag{26}$$

According to the virtual physical system (26), the attacker obtains the malicious sensor measurement $\tilde{y}(t)$ and injects it into the target computing system.

From the viewpoint of the computing system, the malicious sensor measurement $\tilde{y}(t)$ is similar to the response of the physical system to the control input signal $u(t)$ in an attack-free environment owing to the perfect model knowledge of the attacker. Therefore, under the covert attack, the state estimator estimates the state of the attacker's virtual physical system, $x^a(t)$, and the controller calculates the control input signal $u(t)$ to stabilize the virtual state $x^a(t)$ rather than the real physical system state $x(t)$. In other words, the computing system believes that it controls the real physical system, but it controls the virtual physical system of the attacker.

During the injection process, the attacker hijacks the control input signal transmission channel to inject the malicious control input signal $\tilde{u}(t)$. Then, the attacker designs the malicious control input signal $\tilde{u}(t)$, which destabilizes the physical system and injects the input signal $\tilde{u}(t)$ into the physical system. Moreover, the attacker can design the state estimator and the controller to control the physical system in any desired manner.

If the state $x(t)$ is equal to $x^a(t)$ at the start of the attack, there is no residual perturbation, implying perfect stealthiness. If the deviation between $x(t)$ and $x^a(t)$ is small enough that the residual $r(t)$ does not exceed the threshold δ, the residual $r(t)$ in the anomaly detector converges to zero after a small perturbation. Consequently, the anomaly detector does not activate the attack alarm $\Gamma(t)$. Therefore, the stealthiness of the covert attack is guaranteed.

1) Attack Constraints: To implement the covert attack, the attacker must take over the transmission channels for both the sensor measurement and the control input signal. In addition, the attacker can sniff the control input signal to calculate $x^a(t)$ in the virtual physical system (26), which deceives the computing system.

In the deception process, the attacker must already have a precise model of the physical system to construct the virtual physical system (26). If there is a mismatch between the virtual and real physical systems, the malicious sensor measurement $\tilde{y}(t)$ cannot guarantee the stealthiness of the covert attack.

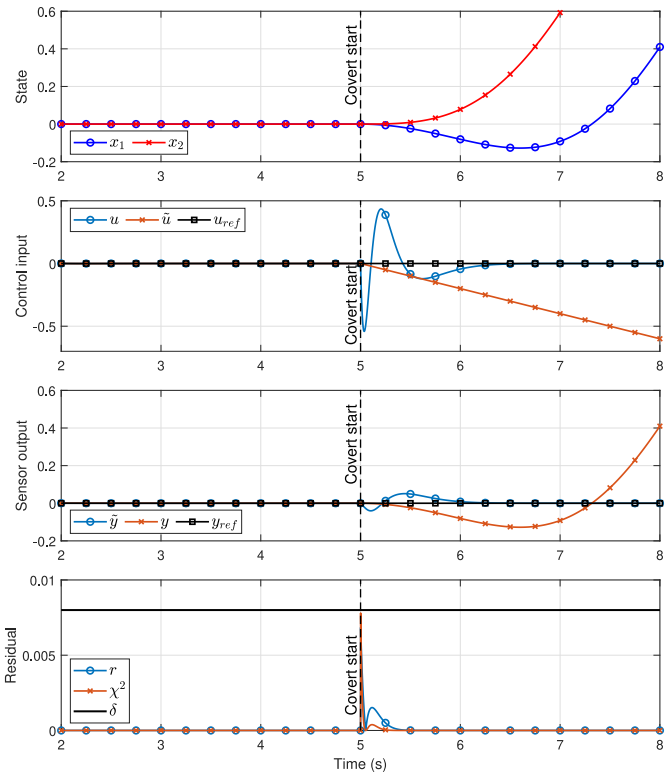


Fig. 18. Numerical example of the covert attack.

In the injection process, in contrast to the ZDA and the PDA, there are no constraints on the physical system before launching the attack. If the attacker wants to control the physical system the way it wants, it is necessary to sniff the sensor measurement $y(t)$ with a proper state estimator and controller to generate the malicious control input signal $\tilde{u}(t)$.

2) Numerical Example: We implement a simulation study of a covert attack against a physical system with the dynamics (18) as shown in Fig. 18. In this simulation, we simultaneously inject the malicious sensor measurement $\tilde{y}(t)$ and the control input signal $\tilde{u}(t)$.

The physical impact of the covert attack illustrated in Fig. 18 is based on the malicious control input signal $\tilde{u}(t)$ in the injection process from the start of the attack at $t = 5$ s. The divergence of physical state $x(t)$ shows that the covert attack renders the physical system unstable.

In the deception process, the computing system functions as if the original control signal $u(t)$ successfully transmitted to the physical system. In this case, a small perturbation of the control input $u(t)$ is triggered by a deviation between the state of the real physical system, $x(t)$, and the state of the attacker $x^a(t)$. The physical system receives the malicious control input signal $\tilde{u}(t)$ via the injection process of the covert attack. In this case, the malicious control input signal $\tilde{u}(t)$ is designed as $\tilde{u}(t) = -0.2(t-5)$ in the simulation. The difference between the control input signal $\tilde{u}(t)$ by the covert attack and the original control input signal $u_{ref}(t)$ renders the physical system unstable.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

The original sensor measurement, $y(t)$, from the real physical system becomes unstable via the injection process of the covert attack. However, as a result of the deception process, the computing system receives the manipulated sensor measurement $\tilde{y}(t)$ from the virtual physical system (26) of the attacker. This deception process causes only a temporary small perturbation by the deviation between states $x(t)$ and $x^a(t)$, as well as in the control input signal $u(t)$. Following the small perturbation at the start of the deception process, the malicious sensor measurement $\tilde{y}(t)$ traces the sensor measurement $y_{ref}(t)$.

We evaluate the stealthiness of the PDA with the residual-based detector and the χ^2 detector. We set the threshold $\delta = 0.008$ to detect the covert attack in two detectors. Because of the state estimate $x^a(t)$ of the state of the attacker in the deception process, the residual $r(t)$ converges to zero after the transient perturbation of the initial state of the virtual physical system $x^a(t)$. As illustrated in Fig. 18, the residual does not exceed the threshold δ. As a result, the residual-based detector does not activate any attack alarm $\Gamma(t)$. In addition, the quantity $g(t)$ of the χ^2 detector (13) does not exceed the threshold δ. Therefore, the χ^2 detector does not activate any attack alarm $\Gamma(t)$ in response to the covert attack. Consequently, the covert attack is stealthy with disruption resources.

3) Improved Attacks: The constraint that the attacker must have a precise model of the physical system renders it difficult for the attacker to launch a covert attack. For relaxation of the covert attack constraint, a data-driven covert attack is proposed in [101], where the attack is designed with a subspace predictive control method, an intercepted control input signal, and the sensor measurement of the network.

4) Detection Strategy: A covert attack detection architecture with an auxiliary system is proposed in [53], where a truthful auxiliary system is equipped in the physical system, and a switched Luenberger observer is adopted to handle the auxiliary system. In [97], a modulation matrix insertion method on the control input signal path, detects both the covert attack and the ZDA.

VIII. Resilient CPS Approaches

In previous sections, we presented an in-depth analysis of existing cyber-physical attacks and the attack detection strategies. However, after attack detection, attack handling and mitigating strategies are required for CPS to provide sustainable services under attacks. Furthermore, there are other situations, such as system faults, that can also render CPS unstable. Therefore, CPS must provide sustainable operations even when CPS face system faults.

In this section, we introduce resilient CPS design approaches against cyber-physical attacks as well as system faults in terms of secure control, network management, and data-driven strategies. In Table VI, the resilient approaches against cyber-physical attacks and system faults are summarized, where data manipulation includes stealthy cyber-physical attacks. We summarize resilient CPS design approaches against cyber-physical threats with proper references as shown in Table VI.

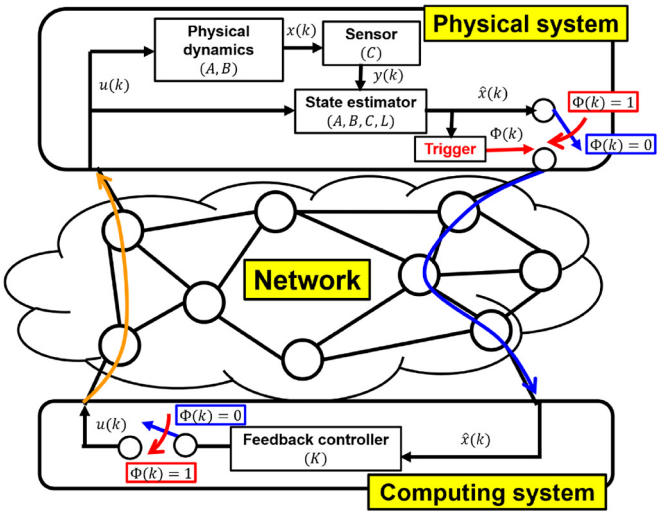


Fig. 19. Overview of event triggered control for CPS.

A. Secure Control Strategy

As computing technology advances, physical systems in CPS have light-weight computation resources. From the computing resources, the physical systems can process the control input signals and sensor measurements, which support various control methods, including aperiodic control and modulation of the sensor measurement and the control input signal. By applying an appropriate control method to CPS, the physical systems become resistant to some known cyber-physical attacks.

1) Event-Triggered Control: Event-triggered control (ETC) is an aperiodic control strategy that transmits the sensor measurement $y(k)$ and the control input signal $u(k)$ when a specific event occurs [135]. In contrast to the periodic control system illustrated in Fig. 3, the ETC system has a state estimator in the physical system as shown in Fig. 19. In ETC, the event is generated by a triggering function related to the stability of the physical state, $x(k)$, such as a Lyapunov function [106], [136]. In most cases, it is formidable for sensors to obtain all the state variables of $x(t)$. Thus, ETC strategies can be applied to CPS that enable the embedding of a state estimator in the physical system to trigger an event based on state estimation $\hat{x}(k)$.

The ETC system exchanges the sensor measurements and the control input signals only when the state estimation $\hat{x}(k)$ incurs an event through violation of the stability condition in the triggering function. In this manner, ETC utilizes the network resources significantly less than a periodic control system [110]. Consequently, because of the reduced usage of network resources, more physical systems can be connected to the network with ETC, compared to the case when the physical systems transmit the sensor measurement $y(t)$ periodically.

From the viewpoint of cyber-physical security, ETC provides advantages for designing resilient CPS. The ETC strategy is well-established for its robustness in the face of unreliable communication channels and packet drops [106]. In addition, control-theoretical stability analysis of physical systems under a DoS attack is demonstrated in [108]

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

TABLE VI
RESILIENT CPS DESIGN APPROACHES AGAINST CYBER-PHYSICAL ATTACKS AND SYSTEM FAULTS

References	Approach		Defense against	Key idea
[106]	Secure control	Event-triggered control (ETC)	DoS attack	Select the controller gain to guarantee the stability by the packet drop probability
[108], [109]	Secure control	Event-triggered control (ETC)	DoS attack	Reduce the packet drop probability with aperiodic packet transmission against random jamming
[110]	Secure control	Event-triggered control (ETC)	Energy consumption	Reduce the communication cost by aperiodic control packet transmission
[98]	Secure control	Coding/Encryption	Data manipulation	Modulate both the control input signal and the sensor measurement with control theoretical matrix operations
[111], [112]	Secure control	Coding/Encryption	Data manipulation	Adopt homomorphic encryption to the control input signal and the sensor measurement
[113]	Secure control	Coding/Encryption	Data manipulation	Reduce the computation complexity of homomorphic encryption for control systems
[114]	Network management	Software-defined networking	DoS	Handle malicious traffic with proactive forwarding rules
[115]	Network management	Software-defined networking	DoS	Filter malicious packet flooding with frequency-based and traffic-based policies
[5], [11]	Network management	Software-defined networking	DoS	Detect topology changes and re-establish the alternative data transmission path on the SDN controller
[116]–[119]	Network management	Software-defined networking	Data manipulation	Sample the packets on the data plane and analyze malicious traffic features
[16]	Network management	Software-defined networking	Data manipulation	Directly inspect attacks on the data plane by utilizing computing resources of SDN switches
[120]	Network management	Co-design of network and control	Network congestion	Schedule network traffic in consideration of task priority and physical system faults
[121]	Network management	Co-design of network and control	Energy consumption	Schedule of multi-hop wireless networks in consideration of energy fairness of each node and control power to satisfy the deadline constraint of packets
[122]	Network management	Co-design of network and control	DoS	Analyze the trade-off between the communication bandwidth and physical resilience against DoS attacks
[123]	Network management	Co-design of network and control	DoS	Exchange control input signals at uniformly distributed random time instants to conceal communication attempts
[124]	Network management	Co-design of network and control	Energy consumption	Adjust the packet transmission rate to reduce network energy consumption while guaranteeing the stability of the physical system under uncertain disturbances
[125]	Network management	Co-design of network and control	Network congestion	Schedule wireless communication nodes and routes packets to guarantee maximum packet arrival time constraints.
[126]	Data-driven strategy	Data-driven fault mitigation	Data manipulation	Combine a nonlinear controller and a neural-network based controller to estimate and to compensate the attacks.
[127]	Data-driven strategy	Data-driven fault mitigation	Data manipulation	Adopt Q-learning with long-short term memory to extract attack features from time-series sensor measurements
[128]	Data-driven strategy	Data-driven fault mitigation	Data manipulation	Adopt reinforcement learning with the transient energy of grid system as a reward
[129]	Data-driven strategy	Data-driven fault mitigation	Data manipulation	Classify cyber-physical attacks with a support vector machine model and a switch controller to recover physical processes
[130]	Data-driven strategy	Data-driven fault mitigation	Data manipulation	Detect cyber-physical attacks and select proper attack mitigation strategies with a machine learning method
[131]	Data-driven strategy	Data-driven fault mitigation	DoS	Learning wireless network conditions and provide adaptive geometric routing
[132]	Data-driven strategy	Data-driven fault mitigation	Network congestion	Schedule of vehicular networks with deep-Q learning that utilizes the network life time as constraints
[133]	Data-driven strategy	Generative adversarial network	Data manipulation	Generate an artificial attack signal to train a machine-learning based anomaly detector
[134]	Data-driven strategy	Generative adversarial network	Data manipulation	Conduct a recovery process by external attacks, where the learning-based attack compensator is trained from an attack signal generator

and [109]. The study in [137] shows the stability guarantee of an ETC strategy in the distributed NCS environment. An ETC controller design method is proposed to provide robustness against the DoS attack in [138].

However, the ETC strategy is vulnerable to disturbances in the physical system. When continuous physical disturbances are forced on a physical system, then the triggering function is activated during the disturbances. Under this disturbance scenario, ETC is equivalent to a fast periodic control system. Consequently, the forced external disturbances utilize more network resources and may cause control performance degradation owing to delays caused by increased network traffic.

2) Coding and Encryption of Control-Related Data: By exploiting disclosure resources, the attacker is enabled to implement advanced stealthy cyber-physical attacks such as the robust ZDA and PDA with an uncertain physical system model. To enhance the security of CPS, it is important to prevent the leaking of control input signals and sensor measurements on the network. As a countermeasure against eavesdropping, encryption and coding techniques for control input signals and the sensor measurements on the network are suggested. However, the conventional encryption techniques illustrated in Fig. 20(a) require complex computation that can cause processing delays in both physical and computing systems. These delays in turn result in a degradation of control

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

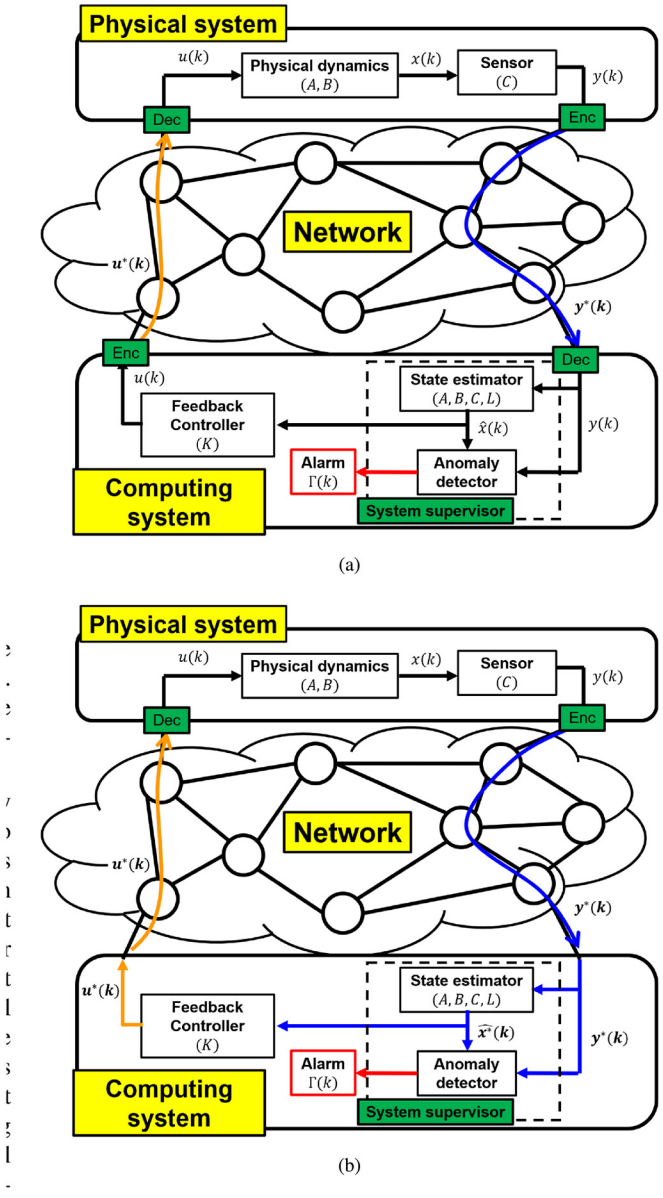


Fig. 20. Coding/encryption of sensor measurements and control input signals on CPS: (a) traditional coding/encryption, (b) homomorphic encryption.

performance and extra power consumption by the physical system. Powerful encryption techniques can be adopted for CPS only when the physical systems have sufficient computing and power resources. Therefore, lightweight coding and encryption methods are required to simultaneously ensure the privacy of CPS and control performance.

In [98], a coding scheme for the control input signal $u(k)$ and the sensor measurement $y(k)$ is proposed to reveal stealthiness in cyber-physical attacks. In contrast to the conventional one-way coding, modulations of both the control input signal $u(k)$ and the sensor measurement $y(k)$ are correlated on the network. The proposed coding scheme restricts the attacker from designing attack signals and provides correction of manipulated control input signals and sensor measurements from the attacks.

The homomorphic encryption technique illustrated in Fig. 20(b) is proposed in [111], which supports mathematical operations directly on the encrypted messages. Compared to the conventional encryption method, this homomorphic encryption approach reduces the computation stages for encryption and decryption while enhancing the privacy of CPS. In conventional encryption methods, as illustrated in Fig. 20(a), both the physical system and the computing system implement encryption and decryption. Homomorphic encryption only requires encryption and decryption on the physical system because the controller can calculate the control input signal $u(k)$ without decrypting the sensor measurement $y(k)$ as illustrated in Fig. 20(b).

In [112], a distributed state observer is proposed with homomorphic encryption, which supports decentralized control in the NCS environment. However, the homomorphic encryption technique requires significantly more computing resources. To overcome the complexity of homomorphic encryption and guarantee the real-time constraint of CPS, a linear homomorphic authenticated encryption (LinHAE) scheme is proposed in [113]. LinHAE supports fast linear operation for encryption to guarantee the real-time control of the physical system. The LinHAE scheme is evaluated in a drone control testbed, where experimental results show that the LinHAE scheme provides real-time attack detection and has a decryption time of within several hundred microseconds.

Encryption methods for CPS and real-time IoT should consider not only the processing delay for encryption/decryption but also hardware limitations, such as memory, battery, and computing power. In [139], the transmission deadline and the encryption latency for computing power-limited in-vehicle networks are analyzed, where the authors evaluate the latency of the conventional AES encryption method in a hardware-in-the loop environment. The evaluation result shows that the worst case latency of the AES algorithm is in the hundreds of microseconds, whereas the maximum allowable latency is less than 10 ms.

Because of the real-time constraints of CPS and IoT devices, encryption techniques must provide high-level security while maintaining low complexity and latency in limited computation resources. The ISO/IEC 29192 standard [140] introduces lightweight encryption mechanisms such as PRESENT [141], CLEFIA [142], and LEA [143], which are suitable for adoption in IoT systems with limited computation resources. PRESENT encrypts data with 64-bit blocks, and the other mechanisms encrypt data with 128-bit blocks. In 2013, the national security agency (NSA) released lightweight block encryption algorithms, including SPECK, which support flexible block sizes for data encryption [144]. A lightweight encryption and authentication methods for smart grid communication are proposed in [145], where communication nodes have limited memory size and computation capacity. Furthermore, various light-weight cryptography strategies, including NIST's upcoming standards [146], can be considered in terms of battery consumption [147].

B. Network Management Strategy

Although the cyber-physical attacks are detected in a control-theoretical manner, the actual recovery of CPS against

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

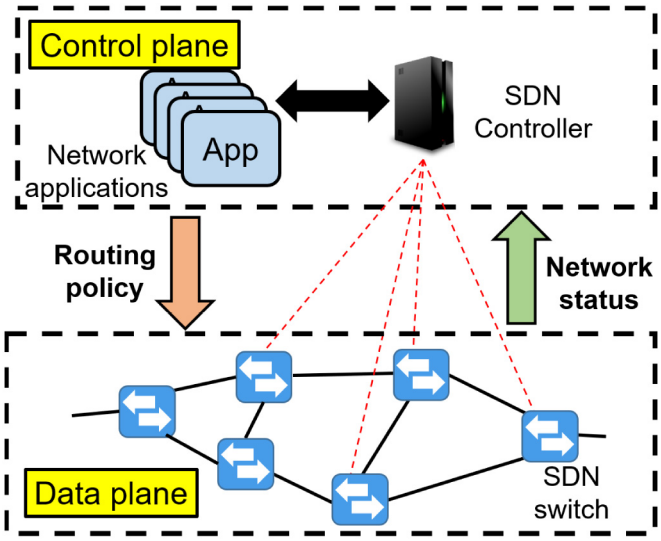


Fig. 21. Structure of software-defined networking.

the attacks is performed on the network, which satisfies deadline constraints from physical systems. Thus, the CPS network should be monitored in real-time and provide flexible network services to handle the attacks. In addition, the stability of the physical system is jeopardized not only by the attack but also by the degradation of network performance; for example, the network should be designed considering both physical aspects and network resources.

1) Software-Defined Networking (SDN): Software-defined networking (SDN) is a new networking paradigm that separates the control plane and the data plane from the conventional network equipment with software, as shown in Fig. 21. In SDN, both the control plane and the data plane are programmable [148]. An SDN network consists of a centralized SDN controller and multiple SDN switches. The SDN controller monitors the network and provides various network services. The SDN switches forward packets based on forwarding rules from the SDN controller. The interconnection between the SDN controller and the SDN switches is supported by a southbound API such as OpenFlow [149].

The SDN controller on the control plane periodically collects network information, including traffic loads for each link from the SDN switches. The SDN controller also installs forwarding rules on each SDN switch on the data plane to provide network services such as a quality of service (QoS) guarantee [120], flexible routing [150], and load balancing [121]. The SDN switches on the data plane transmit the network status of each switch to the centralized SDN controller and forward packets according to the forwarding rules of the SDN controller [151].

An SDN network can mitigate the impact of a DoS attack by malicious traffic flooding with various network applications and SDN switch programming. In [114], a FloodGuard scheme adopts proactive flow rules on SDN switches, which handle malicious traffic causing saturation of the control plane of the network. As a prevention measure against a DoS attack, FloodDefender [115] detects and filters malicious traffic flooding on the control plane. The experimental result shows mitigation of the DoS attack by sustaining CPU utilization under 0.5% and only a few tens of milliseconds of delay to handle the packets.

The SDN network also provides link-failover for network link breakdowns. Link-failover recovers broken feedback control loops between physical systems and computing systems. In [5], an SDN network with link redundancy recovers the control data transmission path under an intentional link breakdown attack in the CBTC environment. The SDN controller searches for an alternative data path, and re-establishes the connection between a train on the rails and a supervision system on the ground. For micro-grid systems, an SDN-based link-failover strategy [11] also provides voltage recovery from link damage between the control center and energy storage systems. The analysis shows resiliency and better recovery speed than other conventional recovery protocols, such as the spanning-tree protocol (STP), which fails in voltage level recovery and rapid STP (RSTP), which requires more time to recover the voltage level.

Traffic monitoring of the SDN network through a southbound API enhances security against false-data injection types of cyber-physical attacks [116]. The SDN controller with powerful computing performance can directly analyze mirrored traffic of the data plane with the IDS because the traffic is transmitted to the SDN controller as well as OpenFlow packets [117] for malware detection. A placement of the traffic sampling system is proposed in [118] to reduce duplicated packets while maintaining security for CPS. On the data plane, programmable SDN switches with additional computing resources can detect network anomalies such as MITM attacks. The distributed SDN switches in the network periodically capture the packets inserted into each switch and analyze the anomalies like the IDS with a previously installed detection engine [116]. If anomalies in the packets are detected, then the SDN switch informs the SDN controller of the attack alarm, and a human analyst takes the appropriate actions [116].

A lightweight countermeasure to detect MITM attacks is proposed in [119], where the features of the OpenFlow response are utilized for various MITM attacks. In [16], an empirical study is carried out on MITM attacks to cause train collisions in the CBTC environment. Here, an SDN switch-based countermeasure is proposed to detect the attempted MITM attacks. The results of the empirical study in [16] show that the proposed countermeasure can operate an auxiliary emergency breaking system within seconds, and can guarantee the resiliency of the CBTC system under attack.

However, the SDN is known to be vulnerable to a distributed DoS (DDoS) attack, where massive communication devices access and request network connections. When a DDoS attack is launched on the SDN network, the control plane of the SDN network becomes vulnerable because the SDN controller and SDN switches should be able to handle requests from massive attacker devices, where notorious traffic for simultaneous connection requests is generated. Consequently, massive malicious network access exhausts the computing and communication resources of the SDN controller and SDN switches, and SDN network management functions cannot operate.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

2) Co-Design of Network and Control: CPS network design should be considered along with the aspects of network saturation [152], energy consumption [153], and the stability of the physical system [154]. Network performance degradation caused by any factor should not cause physical instability [155]. Most cyber-physical attacks involve network delays, which can degrade the performance of the physical system, or render the physical system unstable. In other words, if the detection and recovery strategies operate normally on the network, an attack-induced delay can affect the operation of the physical system. Therefore, to mitigate performance degradation induced by the network in CPS, a significant amount of research on network design has been conducted, such as scheduling [156], [157] and data-rate adaptation [157] along with physical stability [122], [124], [125].

An SDN-based QoS guarantee is proposed in [120], which classifies and schedules traffic considering task priority. The experimental result in [120] shows that the proposed SDN technique handles increased traffic induced by physical faults and provides fault-tolerant control of the physical system. In a multi-hop wireless network environment, the SDN-based traffic scheduling solution in [121] simultaneously considers the energy consumption fairness of network nodes as well as handles urgent packets by CPS anomalies within a deadline.

In [122], a data-rate selection strategy is proposed under DoS attacks, where the authors consider a minimum data rate to guarantee the stability of the physical system based on the DoS attack frequency and the average dwell time. A randomized transmission protocol for NCS is proposed in [123] to protect against DoS attacks. The randomized protocol in [123] renders reactive jamming more difficult to implement for the attacker. Rate adaptation, proposed in [124] for multi-hop wireless networks, ensures the stability of the control system while reducing network energy consumption when a physical perturbation exists. The cross-layer optimization protocol (CLOC) for time-division multiplexing access (TDMA) network scheduling in [125] considers physical dynamics and network constraints, including routing and concurrent transmissions. The CLOC achieves better communication reliability in terms of the maximum allowable transmission interval (MATI) requirement and better control performance than fixed scheduling.

C. Data-Driven Strategy

As CPS become larger and more complex, mathematical modeling of CPS does not reflect the real CPS with high accuracy. The CPS components that are not reflected in the mathematical model can become novel CPS vulnerabilities that can be further exploited by the attacker with stealthiness. Data-driven techniques, which represent complex systems with enormous data and high-performance computing, fill the vulnerability gap between the mathematical model and the real CPS. Furthermore, the combination of control-theoretic and data-driven techniques can provide mitigation and recovery of CPS under cyber-physical attacks.

1) Data-Driven Fault Mitigation: The application layer can acquire various chunks of data from the physical and network layers, such as the physical state and network statistics. From the acquired data, machine learning techniques can provide a secure state estimation against malicious data deception by cyber-physical attacks. Then, the secure estimation mitigates the physical impact of the attack. In addition, the machine learning techniques can guarantee robustness not only against cyber-physical attacks but also against system uncertainties such as an unexpected system fault and battery exhaustion that may cause a violation of CPS safety conditions. To enhance the robustness of CPS against attacks and uncertainties, data-driven strategies can be adopted with appropriate machine learning models in network and physical domains.

For the physical layer, data-driven strategies provide proper actions to mitigate physical impacts from system faults. A hybrid controller scheme based on a neural network and a nonlinear controller for attack estimation and compensation is proposed in [126], where the learning-based attack estimator works in an online manner and the nonlinear controller compensates for an attack in real-time. To enhance the robustness of an autonomous vehicle against cyber-physical attacks, reinforcement learning techniques considering physical dynamics are being investigated in [127]. The proposed technique mitigates the deviation between the ground truth location and the estimated location of the vehicle against an attacker that modifies inter-beacon information of the vehicles. A reinforcement learning-based smart grid recovery method is proposed in [128], where transient energy is utilized as a reward for reinforcement learning. Evaluation results of the proposed recovery method show that current fluctuation by cyber attack is mitigated, and the mitigation performance is enhanced compared to the conventional deep-Q network (DQN) based leaning method.

A physical process-aware cyber-physical attack detection and mitigation method is proposed in [129], which learns the types of attacks with a support vector machine (SVM) model and switches the controller to a recovery strategy against the attack. The experimental result is provided to categorize various attacks in normal and disturbance conditions and to satisfy the stability of the physical process against controller attacks on the hardware-in-the-loop testbed for a chemical plant. To mitigate time-delay attacks on a power grid, the authors in [130] present a machine learning-based attack handling method with two-state classifiers and two mitigation techniques. When the violation of stability or safety conditions is detected by machine learning methods, the proposed method handles the impact of the attack with a two-stage mitigation strategy: control gain tuning and load shedding, where the load shedding is executed only if the gain tuning cannot guarantee stability.

For the network layer, data-driven strategies provide reliable networking under attacks and guarantee safety conditions under system constraints. A secure routing protocol against jamming attacks on a carrier sense multiple access with collision avoidance (CSMA/CA) network is proposed in [131] with a machine learning technique, where the proposed geometric routing protocol is adaptive to network conditions and provides better network throughput than other routing protocols under jamming attacks. A Q-learning technique with radio channel

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

state information is proposed in [158], which offers radio layer authentication to enhance resistance against spoofing attacks.

The authentication performance by the proposed learning techniques [158] is compared with the fixed threshold method in terms of the average authentication error rate and the receiver utility. In [132], a network scheduling method addressing both the safety and QoS level is proposed with a deep Q-network (DQN) model, which reduces the average request delay related to safety and enhances the completed request percentage related to the QoS level while extending the network lifetime of a battery-powered vehicular network.

The data-driven approaches require massive data for CPS operations under normal and attack situations. However, in the real CPS environment, it is difficult to obtain massive data for abnormal behavior of CPS because the types of cyber-physical attacks are limited. Imbalance between normal and abnormal CPS operation data causes over-fitting or under-fitting problems in establishing attack decision criteria, which degrades the attack detection performance and does not guarantee detection accuracy against a novel cyber-physical attack.

2) Generative Adversarial Networks (GAN): In [159], an anomaly detection dataset-generation research is conducted for any types of attacks, including command modification and replay attacks. The study is composed of attack selection, attack deployment, traffic capture, and feature selection. The generated datasets by target attack features are used to train other data-driven anomaly detectors for heterogeneous CPS. However, it is not a simple task to collect sufficient cyber-physical attack samples for training machine learning models for attack identification. Lack of attack signal data for training provides poor performance in terms of accuracy of attack classification, such as detection miss and false-positive detection.

Generative adversarial networks (GAN) [160] can be used to generate a dataset of various cyber-physical attacks, where the generated dataset can be used to train a machine learning model for physical anomaly detection. Fig. 22 illustrates the structure of GAN, which consist of two different neural networks known as the generator and the discriminator.

The generator creates a malicious signal with a random noise z and a differentiable generator function G, where the aim of the generator is to deceive the discriminator. The discriminator with a differentiable discriminator function D is responsible for classifying a legitimate signal by a dataset and a fake signal by the generator. The discriminator assigns the label âĂIJtrueâĂİ to the legitimate signal, and âĂIJfalseâĂİ to the fake signal by the generator. A classification error of the discriminator is backpropagated to both the generator and the discriminator for training, and these neural networks are updated by various training methods [161] including the stochastic gradient. This training is conducted until the discriminator no longer differentiates between the legitimate signal and the fake signal generated by the generator.

To resolve the issue of a lack of fault data from industrial machines, the authors in [133] propose a novel fault detection method with GAN, where the generator in GAN artificially creates the machine fault data that resembles the original fault signal of the real machine. The proposed fault detection framework is trained by both real and artificial fault data. An experimental result on a testbed shows the fault detection accuracy of the proposed framework, which is higher than other conventional machine learning models.

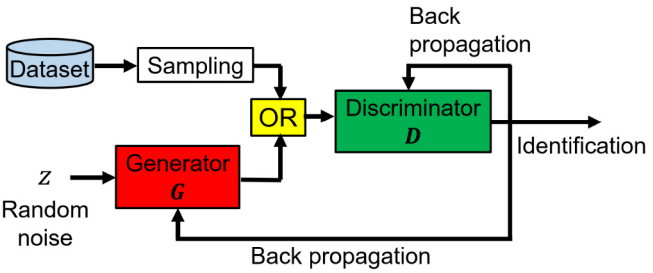


Fig. 22. Structure of the generative adversarial networks.

In contrast to the conventional GAN that only provides attack detection, conditional GAN (CGAN) for battery management systems are proposed in [134], which simultaneously provide detection and recovery functions of the physical system against attacks. The CGAN capture and learn the time-series dynamic behavior of the system, where a certain part of the state is utilized as conditional data to train both the generator and the discriminator. When the attack is detected, the generator estimates the correct state of the physical system with the conditional data that are part of the sampled state. From the estimation of the generator, the controller conducts the recovery process of the physical system after the attacks. In [162], a GAN framework for multi-variable anomaly detection is proposed, where the long-short term memory RNN (LSTM-RNN) model is adopted for both the generator and the discriminator. A reconstruction score that evaluates the similarity of signals between a test dataset and the generator and a discrimination score that evaluates anomaly detection on the discriminator are exploited to determine anomalies in CPS. The authors in [162] evaluate the proposed GAN framework on a water treatment testbed.

D. Resilient CPS Example 1: Inverted Pendulum Control Under DoS Attack

Resilient CPS provide real-time performance recovery under cyber-physical attacks, where the total time required for attack detection and recovery must remain within the deadline. We present an illustrative example of control performance recovery with a resilient CPS strategy against the DoS attack, which shows the CPS deadline constraint and state transition by the recovery.

1) Physical System: We consider a linearized inverted pendulum example [163], which is one of the typical physical systems to evaluate control performance. The dynamics of a second-order linearized inverted pendulum is given by

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ \frac{(m+M)g}{Ml} & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ -\frac{1}{Ml} \end{bmatrix} u(t)$$
$$u(t) = -K\mathbf{x}(t),$$

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

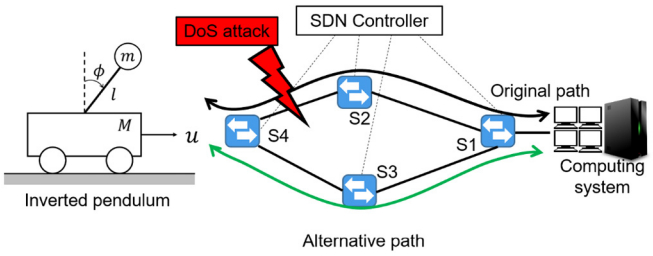


Fig. 23. DoS attack recovery scenario.

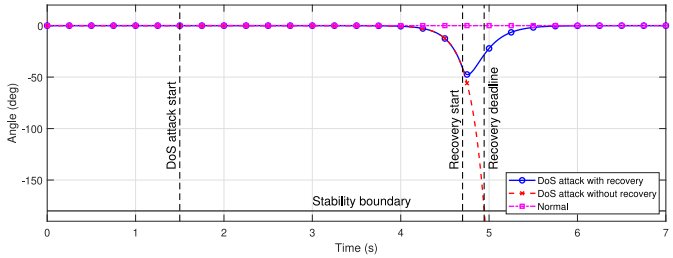


Fig. 24. Real-time attack recovery of the inverted pendulum.

where $\mathbf{x}(t) = [\phi(t)\ \dot{\phi}(t)]^T$ is the state vector of the inverted pendulum, $\phi(t)$ is the pendulum angle, m is the pendulum mass, M is the cart mass, l is the pendulum length, and g is gravitational acceleration. The goal of control is to regulate the angle $\phi(t)$ to zero. In addition, the inverted pendulum has an angle $\phi(t)$ within the range of $[-180°, 180°]$. Therefore, the inverted pendulum is unstable if the angle $\phi(t)$ moves beyond the angle range.

2) Network Structure: We consider the network recovery example in [100], where the network topology consists of an SDN controller and four SDN switches, as illustrated in Fig. 23. The computing system and physical system periodically exchange the state $x(t)$, rather than the sensor measurement $y(t)$ to simplify the state estimation process, and control input signals $u(t)$ through the upper path $S1-S2-S4$ with the general UDP protocol. The SDN controller and SDN switches communicate with the OpenFlow protocol to manage the entire network and report network status. The role of the SDN network is to set an alternative data path $S1-S3-S4$ between the computing system and the physical system when the original data path suffers the DoS attack. We assume that the DoS attack is reported by a certain method to the SDN controller, such as a safety checker of the computing system or network status monitoring of SDN switches, and that the SDN control plane is not attacked.

3) Recovery From the DoS Attack: We implement a simulation study of inverted pendulum recovery against the DoS attack as shown in Fig. 24. In this simulation, we launch the DoS attack at the attack start time $t_s = 1.5$ s on the network with 3.2 s of attack duration, and assume that the recovery of the inverted pendulum is started at the attack recovery start time $t_r = 4.7$ s. In a networking viewpoint, the UDP packets, including the state $x(t)$ and the control input signal $u(t)$ on the original data path are dropped under the DoS attack, and the physical system becomes an open-loop system holding the previous control input signal $u(t_s)$. At the recovery start time t_r, the SDN controller completes establishing an alternative data transmission path, and the exchange of the state $x(t)$ and the control input signal $u(t)$ is resumed.

According to the simulation result shown in Fig. 24, the angle of the inverted pendulum exceeds the range at the recovery deadline $t_d = 4.94$ s. Therefore, the attack recovery process must be executed within a real-time constraint of $t_d - t_s = 3.44$ s. Consequently, in this example, a resilient CPS strategy must be designed to guarantee the execution time, including attack detection and network recovery, within the real-time constraint.

E. Resilient CPS Example 2: Communication-Based Train Control (CBTC) Under Controller Attack

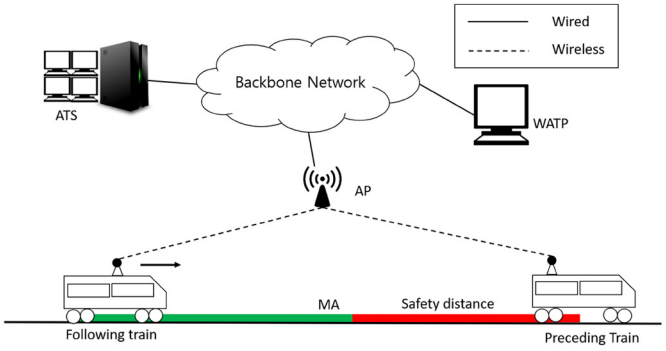
The CBTC system is one of the practical CPS that can be remotely controlled through the wireless networks in real-time. In our previous studies [5], [16], we analyze the impact of MITM attacks on the CBTC system, and design the countermeasure using the SDN technique. Furthermore, we evaluate the countermeasure in a realistic testbed that consists of commercial train control software, real track information, and a railway-specific standard communication protocol [5]. We present an example of safety guarantee against a controller attack through the wireless network, where the proposed mechanism [16] provides detection and resiliency within the deadline.

1) CBTC System: The CBTC system consists of trains on the railway and control facilities on the ground, such as automatic train supervision (ATS) and wayside automatic train protection (WATP), as illustrated in Fig. 25(a). Trains and ground facilities interact through railway specific wireless communication such as LTE-R or IEEE 802.11 wireless networks. The trains periodically transmit sensor measurements to the ground facilities, and the ground facilities provide control commands to the trains in a specific packet format. Therefore, the trains on the railway and ground facilities establish feedback control loops through the wireless network.

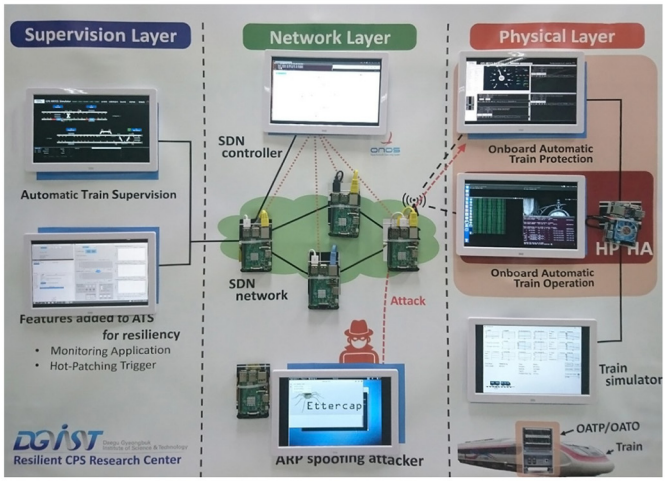
The control commands include the movement authority (MA) that determines the maximum allowable travel distance and the most restrictive speed profile (MRSP) that specifies the speed limit of the trains in a certain section. Furthermore, sensor measurements from the trains provide the current speed measured by odometers and the current vehicle position measured by GPS or tags on the railway. In the ground facilities, the MA and MRSP are calculated by the sensor measurements, specifications of trains, and geographic information. Because of centralized control through wireless communications, CBTC systems provide shorter travel times than conventional train control systems, while guaranteeing a safe distance between trains to avoid train collision accidents.

2) Network Structure: In our testbed, we construct an SDN network including wired and wireless networks as illustrated in Fig. 25(b), where wired networks and wireless networks correspond to the backbone networks of ground facilities and radio-access communication between trains and ground facilities, respectively. The wired networks consist of four SDN

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

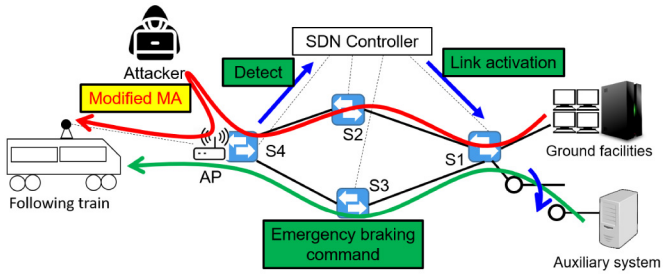


(a)

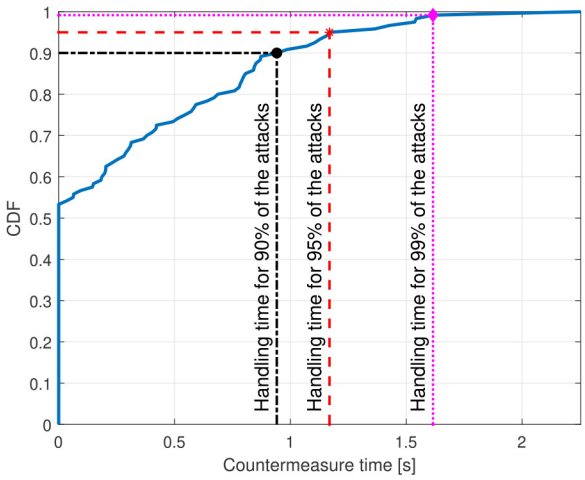


(b)

Fig. 25. Communication-based train control (CBTC) system: (a) illustration of CBTC operation, (b) the testbed.



(a)



(b)

Fig. 26. Resilient CPS architecture in CBTC systems: (a) proposed resilient CPS architecture, (b) countermeasure time of the proposed structure.

switches implemented by Raspberry Pi 3s. One of the SDN switches includes an IEEE 802.11 wireless network access point (AP) for the train software to connect to the network.

The SDN switches periodically detect malicious behavior with packet monitoring software. If a certain cyber-physical attack is detected by the packet monitoring software, the SDN switches report the attack to the SDN controller. Then, the SDN controller provides a resilient solution on the SDN network in real-time.

3) Recovery From the Controller Attack: In our testbed, we evaluate the performance of our resilient CPS architecture against the controller attack through the wireless network. We consider an attack scenario with two trains on the railway and an attack device as follows:

1) The preceding train stops at a certain location, and the following train must maintain a safe distance to avoid a train collision accident. We set the maximum speed limit of the trains at 22.2 m/s and the safety distance between two trains at 260 m.
2) The attack device attempts ARP spoofing between the ground facility and the following train, which steals control packets from the ground facilities, as illustrated in Fig. 26(a).
3) MA information in the control packets is modified by the attack device and forwarded to the following train. The modified MA exceeds the location of the preceding train.
4) After receiving the modified MA information, the following train is authorized to move beyond the location of the proceeding train, which results in a collision accident.

In a legitimate situation, the following train stops at a safe distance behind the preceding train. However, because of the MA of manipulated control command, the following train incurs the collision accident.

To avoid the train collision in the attack scenario, we adopt the SDN network and an auxiliary system, and evaluate the performance of the proposed resilient CPS architecture [16] in our testbed. As illustrated in Fig. 26(a), the proposed CPS architecture consists of an SDN controller, SDN switches, and an auxiliary system to provide emergency braking to the compromised train. In an attack-free environment, the connection between the SDN network and the auxiliary system is deactivated, and hence the attack device cannot search for and damage the auxiliary system before the ARP spoofing. In the proposed CPS architecture, the SDN switches periodically detect cyber-physical attacks with packet monitoring software. When a cyber-physical attack is detected, the SDN switches notify the SDN controller of the attack. The SDN controller establishes the connection of the auxiliary system. Finally, the

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

braking command is transmitted to the following train, and the following train attempts emergency braking.

In the attack scenario, the emergency braking deadline is calculated as 11.7 s from the maximum train speed and the safety distance. To evaluate the performance of the proposed CPS architecture, we measure the countermeasure time, which is defined as the time interval from the instant the first modified packet arrives on the following train until the arrival of the train emergency stop packet. Fig. 26(b) shows the countermeasure time of the proposed CPS architecture. In more than half cases, the proposed CPS architecture neutralizes the cyber-physical attack without any impact from the attack device. Furthermore, even in the worst case, the proposed CPS architecture handles the controller attack within 2.25 s. From the experiment, we can verify that the proposed CPS architecture guarantees the resiliency of CPS in real-time.

IX. Potential Research Directions

The goal of resilient CPS is to guarantee the stability of physical systems through proper recovery strategies after attack detection. The recovery strategies should be designed in consideration of the deadline constraint, limited resources, and interaction of each CPS layer illustrated in Fig. 3. This section presents relevant future research directions for resilient CPS design.

A. Resilience for Control

Most results for control-theoretical cyber-physical security focus on attack detection. However, consideration of a time-to-detect metric as a detection strategy is insufficient in various cyber-physical security research. Because the physical systems operate in real-time, the impact of the attack on the stability of the physical system may vary depending on the dynamics of the physical system. Therefore, the attack detection method should be designed with consideration of the attack detection time. The attack must be detected before the physical system deviates from the recoverable region because of the attack.

Even after an attack is detected, the attacker can launch attacks persistently as long as the attacker's network connection is maintained. In addition, an intelligent attacker can change its attack location after noticing that the computing system has detected the attack. Most studies on control-theoretical cyber-physical security focus on attack detection, and there is a little research on reactive attack mitigation. Therefore, it is of great importance to isolate the attacker on the network in real-time, on top of attack detection strategies. An attacker isolation strategy can be implemented in the network layer of CPS with recognition of the attacker location from the network status. Then, various network access control and authorization methods can cut off the communication link of the attack device, which will neutralize the attack.

After attacker isolation, the damaged physical system must be recovered for sustainable CPS operation to guarantee the stability of the physical system. In addition, network path recovery is needed because the existing data transmission path may be destroyed because of attacker isolation. Therefore, it is required to design a recovery strategy that simultaneously considers the control-theoretical and communication networking aspects of CPS. In the control-theoretical approach, the state estimator should estimate the state of the physical system, and the computing system re-configures the feedback controller to mitigate the impact of the attack and recover the physical system safely. Furthermore, the computing system should provide the deadline constraint for the network to recover the communication path between the physical system and the computing system. At the same time, in the network domain, resilient network management methods should search for and re-establish an alternative data transmission path within the deadline constraint from the control-theoretical analysis.

The three processes, i.e., attack detection, attacker isolation, and system recovery, should be performed sequentially in real-time. In these processes, the knowledge of the physical system gives the deadline constraint. Also, the network is responsible for attack isolation and data path recovery within a given deadline constraint to guarantee the stability of the physical system. In this manner, co-design of the network and control is required for the resilience of CPS under cyber-physical attacks.

B. Resource Management of Security Mechanisms

Most of the existing cyber-physical security research requires embedding software for specific attack detection, adopting novel network protocols, or extending the conventional system with an auxiliary system. However, the introduction of additional security mechanisms consumes computing, networking, and power resources. Excessive use of resources for security may adversely affect the safety and control performance of physical systems, particularly when the resources are limited. Therefore, resource management is crucial to guarantee the physical stability of CPS.

Recent machine learning-based cyber-physical security techniques require very high computing performance. However, the computing resources used to detect and mitigate cyber-physical attacks are limited because of hardware constraints of the computing system. Multiple physical systems can be connected to the computing system in CPS. If most of the computing resources are used for the attack detection task of a certain physical system, the stability of other physical systems under normal operation may be compromised due to the increased processing delay in the computing system. Therefore, the introduction of attack detection and mitigation techniques with high computational complexity should be avoided to satisfy the overall real-time constraint of CPS.

A network resource is related to network congestion that causes a transmission delay when multiple physical systems coexist. Certain network recovery protocols generate significant network overhead, which disturbs the transmission of the normal packets even without any malicious users. Furthermore, network resources such as the transmission rate of sensor measurements can be reallocated to an attacked physical system for recovery of control performance when a cyber-physical attack is launched. However, excessive resource reallocation can render the other physical systems unstable

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

because insufficient network resources caused by the reallocation violate the maximum allowable transmission interval (MATI) constraint of the other physical systems. To resolve the network resource allocation problem, a real-time network monitoring technique is needed, such as an SDN controller. QoS management and adaptive transmission rate optimization for each physical system with the network status are required, which can prevent physical systems from being unstable owing to an increased transmission delay.

In battery-powered CPS such as UAV and industrial sensors, the energy constraint is also a critical resource, which is closely related to the lifetime of CPS. In the meantime, there is a trade-off between the CPS lifetime and the sampling rate of sensors that determines the control performance. When the physical systems frequently collect the sensor measurements, the control performance is improved while consuming more energy. In terms of the network layer, CPS with wireless sensor networks should consider the network lifetime due to battery constraints on communication nodes [164]. There is a trade-off between redundant transmission and energy consumption in the entire network [165], in that redundant transmissions enhance reliability with increased energy consumption while reducing the network lifetime [166]. Also, transmission fairness is an important issue for the entire network lifetime because nodes with excessive traffic consume more battery power. Therefore, an energy management technique that considers both the control performance and the network reliability is required to extend the CPS lifetime.

C. Resilient Architecture Design

To design a resilient CPS architecture, various approaches illustrated in Table VI should be adopted with consideration of the CPS structure as shown in Fig. 3. In the physical layer, if the physical system does not have any unstable poles, there is no need to embed the detection strategy against the PDA. In the network layer, if the wireless network protocol is adopted for CPS, signal interference for each physical system needs to be additionally considered. Furthermore, the adopted approaches should be compatible with each other. For example, because the ETC strategy non-periodically generates sensor measurement packets, the network scheduling technique in TDMA networks is not suitable for ETC.

From Section VI to VIII, we present detection and mitigation strategies against various cyber-physical attacks. In real-world scenarios, integration of the detection and mitigation strategies is required. In particular, these strategies should be properly adopted according to the specific properties of each CPS domain. For example, industrial sensors with restricted computational resources cannot adopt the ETC strategy because the computing power of the sensors has limited programmability.

To enhance the overall CPS security level, cross-layer security between the CPS layers illustrated in Fig. 3 should be considered. In Section V, we only consider physics-based cyber-physical attack detectors. However, a resilient CPS architecture includes both designs for physical systems and networks in real CPS environments because each security component can be designed to complement each other. For example, when a cyber-physical attack with unknown network vulnerabilities is launched, the physics-based detector can detect anomalies and provide an appropriate recovery process against the attack. In contrast, in the case of a covert attack that cannot be detected by the conventional anomaly detector, the network IDS can reveal the covert attack by monitoring traffic and network behavior.

X. Summary and Lessons Learned

In this survey, we provide a comprehensive analysis of cyber-physical attacks, detection methods with physical system knowledge, and resilient CPS approaches. The main focus of this survey is that the primary objective of cyber-physical security is to ensure the stability of the physical systems under cyber-physical attacks. To represent the behavior of physical systems, we adopt the basic control theory throughout all sections, and emphasize the stability of the physical systems. In this section, we summarize the contents of this survey, discuss the benefit of the introduction of control theory on cyber-physical security, and provide lessons learned in a qualitative manner.

A. Basics on Control Theory

Cyber-physical attacks can affect and disrupt physical systems. Hence, the analysis of the cyber-physical attacks should show their impacts in physical ways. Therefore, we provide basic knowledge of control theory for general network engineers to understand the behavior of the physical systems of CPS in Section III. We mainly focus on three components for controlling the physical systems: the physical system, the state estimator, and the controller.

1) Physical System: The main focus of Section III is modeling of the physical systems that represents the behavior of the physical systems. We consider single-input single-output linear time-invariant (SISO-LTI) models for the physical systems.

To represent the input-output relationship of the physical systems, we introduce two mathematical system description methods: the transfer function and the state-space equation. To show the behavior of the physical systems in the time domain, we mainly consider the state-space equation in this survey.

From these mathematical system models, we derive four physical system characteristics: zero and pole dynamics, stability, controllability, and observability. These characteristics are essential to controlling the physical system as intended by the user.

2) State Estimator: The state estimator is an algorithm to estimate an unmeasured physical state with the physical system model and sensor measurement, such as the Kalman filter and the Luenberger observer. To estimate the unmeasured state, the physical system must satisfy the observability condition. In this survey, we consider the Luenberger observer as the state estimator.

3) Controller: The controller generates control input signals to operate the physical systems as intended by the user.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

A proper physical system model and the controllability condition are essential to design the controller. In this survey, we consider the state-feedback controller with a state estimator.

B. Cyber-Physical System Description

To handle the CPS security issues, we introduce hierarchical CPS structure consisting of the physical layer, the network layer, and the application layer. The physical layer represents the physical systems in the real world, such as autonomous vehicles and manufacturing facilities, which can be described by mathematical models. The network layer represents wired/wireless networks, including various industrial network protocols, which creates feedback loops between computing systems and physical systems. The application layer represents the computing systems in the cyber world, which monitors the physical systems and provides control services.

We consider CPS as NCS from the control-theoretical viewpoint. NCS consist of the physical system, the network, and the computing system, where each component corresponds to each layer of the hierarchical CPS structure. The physical system is considered a SISO-LTI system in the continuous time domain. Because of the discrete-time data transmission through the network, the computing system must provide control input signals in a discrete manner. For the computing system, we derive a physical system model, and design the controller and the observer in the discrete-time domain.

C. Taxonomy of Cyber-Physical Attacks

Cyber-physical attacks are a type of network attacks that destabilize physical systems. Therefore, cyber-physical attacks should be classified according to their impact on physical systems. We provide the taxonomy of well-established cyber-physical attacks with three perspectives: attack space, attack location, and stealthiness.

1) Attack Space: The attack space consists of three dimensions: system knowledge, disclosure resources, and disruption resources. The system knowledge represents the level of the physical system knowledge revealed to the attacker when the attacker designs an attack. The system knowledge requirement is proportional to the difficulty of designing an attack and detecting it on the computing systems. The purpose of disclosure resources is to provide information to the attacker to generate the attack. Disclosure resources can supplement the lack of system knowledge for generating sophisticated attacks. Disruption resources are related to the damage of cyber-physical attacks on the physical systems.

2) Attack Location: The attack location is a type of data modification manipulated by the attacker on the network, where the attacker can manipulate two types of the data: sensor measurements and control input signals. Therefore, there are three types of the cyber-physical attacks: sensor attacks that manipulate sensor measurements, controller attacks that manipulate control input signals, and combined attacks that manipulate both sensor measurements and control input signals simultaneously. From the viewpoint of control theory, data manipulation of a cyber-physical attack is represented as the addition of an attack signal to legitimate data.

3) Stealthiness: Stealthiness is a property under which an attack is not detected by an anomaly detector based on the physical system model. A cyber-physical attack with stealthiness can destroy the physical system while avoiding an attack detection alarm on the anomaly detector. To ensure the stealthiness, the attack signal must satisfy the stealthiness constraints imposed by the dynamics of the physical systems. Therefore, most cyber-physical attacks with a high-level of system knowledge have stealthiness because the atacker with the high-level system knowledge can satisfy the stealthiness constraints.

D. General Anomaly Detection Strategies

We provide anomaly detection strategies with system characteristics, sensor measurements, and control input signals. We consider three types of anomaly detection strategies: model-based detection, statistical detection, and learning-based detection.

1) Model-Based Detection: Model-based detection is based on the state estimation of the physical system. From the physical system knowledge, sensor measurements, and control input signals, the state estimator on the computing system can predict the physical state of the next time step. If the predicted state violates safety conditions or has differences in sensor measurements, we regard it as an anomaly of the physical systems.

The safety checker confirms the safety of the physical systems with state prediction. If the state prediction or sensor measurement deviates from a normal operation region, the safety checker will generate an alarm for anomaly.

The residual between the sensor measurements and the state estimation is a detection metric for anomalies. The residual is calculated as the deviation between sensor measurements from the network and estimated sensor measurements from the state estimator. If the residual exceeds a certain threshold, the residual-based detector generates an alarm for anomaly detection.

2) Statistical Detection: The basis of statistical detection is that the attack signals generated by the attacker cannot perfectly imitate the probabilistic nature of sensor noises. A statistical detector extracts abnormal behaviors of probabilistic properties from network delays and sensor noises. However, statistical detectors are vulnerable to replay attacks because the recorded signals of the attacker are not artificially generated, and include probabilistic properties of CPS.

The χ^2 detector is widely used to detect DoS attacks and general sensor attacks. In the χ^2 detector, the noises are considered independent and identically distributed zero-mean normal random variables for each time step. The KL divergence-based detector is another typical statistical detector. The KL divergence compares the probability distribution of benign sensor measurements and malicious attack signals.

3) Learning-Based Detection: Machine learning has recently been used for modeling complex physical systems. Distinguishing system faults and cyber-physical attacks on

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

computing systems becomes feasible with the acquired model from machine learning.

Generally, supervised learning techniques are mainly used to detect cyber-physical attacks by training the behaviors of physical systems under normal operation and cyber-physical attacks, respectively. Furthermore, unsupervised learning can enhance the cyber-physical attack detection performance by clustering and extracting abnormal behaviors of the physical systems.

E. Analysis of Cyber-Physical Attacks

We analyze five well-established cyber-physical attacks: the denial-of-service (DoS) attack, the replay attack, the zero-dynamics attack (ZDA), the pole-dynamics attack (PDA), and the covert attack. For these attacks, we provide a numerical analysis of attack signals with physical system dynamics. In particular, some cyber-physical attacks require a high-level of physical system knowledge, such as the ZDA, the PDA, and the covert attack. For these attacks, we show the stealthiness property in a mathematical way.

For each attack, we provide the attack constraints from the viewpoint of the attacker. By numerical examples, we present the physical impact of the cyber-physical attacks and the response of the anomaly detectors, which shows the stealthiness of the attacks. Furthermore, we provide advances and detection strategies for each cyber-physical attack.

1) Attack Constraints: From the viewpoint of the attacker, we provide the constraints for the attacker to generate each cyber-physical attack. The attack constraints include the attack location and system knowledge for the attack generation.

2) Numerical Examples: For each cyber-physical attack, we provide numerical examples to show the physical impact and stealthiness. In each numerical example, we show the states, control input signals, sensor measurements, and responses of the anomaly detectors under normal operation and under cyber-physical attacks. Numerical examples show that the physical state diverges to infinity because of cyber-physical attacks, which means disruption of the physical systems. In addition, we show that the typical cyber-physical attacks are stealthy with respect to the residual-based detector and the χ^2 detector, with the exception of the DoS attack.

3) Improved Attacks: There are two directions for cyber-physical attack improvement: exploring novel vulnerabilities and relaxing attack constraints. Exploring novel vulnerabilities aims to develop a novel cyber-physical attack with stealthiness. A cyber-physical attack with a novel network protocol vulnerability is not detected by conventional network intrusion detection systems. Furthermore, a cyber-physical attack containing an unknown control-theoretical vulnerability is not detected by the anomaly detectors introduced in Section VI.

Realistically, designing stealthy cyber-physical attacks is very hard because it is formidable for the attacker to acquire a precise model of the physical systems. By relaxing attack constraints, improved cyber-physical attacks can be realized. For example, a robust ZDA and a robust PDA enable the attacker to generate the attack signals without precise physical system models.

4) Detection Strategies: In particular, we show that typical cyber-physical attacks, except the DoS attack, have stealthiness against conventional physics-based anomaly detectors. To detect these cyber-physical attacks, we introduce detection strategies for each attack. In the case of a DoS attack, the detection methods do not contain the physical system knowledge because the DoS attack cannot require the physical system dynamics. However, most detection strategies for stealthy attacks utilize physical system knowledge and the mechanisms of each attack. As necessary, some detection methods require auxiliary hardware to detect a certain cyber-physical attack.

F. Resilient CPS Approaches

Security of CPS not only includes attack detection, but also attack handling and mitigation to ensure the stability of the physical systems under cyber-physical attacks. Therefore, we provide resilient CPS design approaches in terms of control, network, and data-driven strategies against system faults and cyber-physical attacks.

1) Secure Control: As computing technology advances, the physical systems have more computation resources available to support aperiodic system controls and data encryption. The use of computing resources on physical systems provides resilience against cyber-physical attacks.

Event-triggered control (ETC) is an aperiodic control technique. The physical systems with ETC determine the need for control and then request control input signals from the computing systems. These aperiodic control properties provide robustness against packet drops by DoS attackers. However, the ETC physical systems are vulnerable to externally imposed disturbances. The disturbance on the ETC physical systems degrades not only control performance but also network performance by utilizing network resources.

The computing resources can be used for coding and encryption of sensor measurements and control input signals. The distorted sensor measurements and control input signals by proper coding methods can restrict the design of the cyber-physical attack signals.

However, the encryption techniques that provide high-level security require complex computations. Processing delays for the encryption may destabilize the physical systems by violating the real-time constraints of CPS. Thus, light-weight encryption standards can be adopted for the physical system to strictly ensure the real-time constraints while enhancing the security levels.

2) Network Management: In resilient CPS, networks are responsible for handling cyber-physical attacks. A software-defined networking (SDN) can provide resilient functions against cyber-physical attacks in real-time. Furthermore, co-design of network and control is required to support the resilient functions and to ensure the stability of the physical systems.

Software-defined networking (SDN) is a novel network paradigm that separates the control plane and the data plane.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

SDN devices on the data plane report abnormal behavior of the attackers to a centralized SDN controller. Then, the SDN controller on the control plane manages separated network devices on the data plane and provides various resilient functions to the networks, such as link-failover to recover the broken feedback loops between the physical systems and the computing systems. The SDN networks are vulnerable to distributed DoS (DDoS) attacks targeting the connection between the control plane and the data plane. The DDoS attack exhausts the network and computing resources of the SDN controller and the SDN devices, resulting in the shutdown of the SDN network.

In the recovery process, the CPS networks must support various resilient CPS functions. Therefore, the CPS network design must consider both network properties and physical stability simultaneously. Network scheduling and data rate adaptation with consideration of the physical systems can mitigate performance degradation of the physical systems.

3) Data-Driven Strategies: Machine-learning techniques can mitigate the impact of cyber-physical attacks. By training cyber-physical attacks, the machine-learning techniques provide compensation for the impact of the cyber-physical attacks, recovering the control performance of the physical systems. Machine-learning techniques can also be applied to networks. However, the machine-learning based mitigation methods usually require massive data of CPS operations under attack situations, where the lack of abnormal CPS behavior data may cause over-fitting or under-fitting problems. The generative adversarial networks (GAN) are machine-learning techniques that can resolve the lack of fault data on CPS.

4) Resilient CPS Examples: To highlight the effect of resilient CPS design, we provide two examples of cyber-physical system recovery: inverted pendulum recovery against a DoS attack and an emergency braking on a CBTC system against a controller attack. The first example provides a resilient CPS design from a control-theoretical viewpoint. In addition, the second example provides a resilient CPS design in a realistic scenario.

The first example is a recovery scenario against a DoS attack. In this example, we consider the DoS attack targeting NCS to sustain the stability of an inverted pendulum. From the numerical simulation, we derive a deadline constraint for the network recovery against the DoS attack. Finally, we show the timely recovery of the inverted pendulum under DoS attack.

The second example is an emergency braking scenario against a controller attack on a CBTC system [16]. In this example, we consider the controller attack that causes a train collision accident. From the attack scenario, we derive a deadline constraint to avoid the train collision, and show the resiliency of an SDN-based attack handling method.

G. Potential Research Directions

The goal of resilient CPS is to guarantee the stability of the physical systems under various cyber-physical attacks and system faults. To advance the resiliency of CPS, we propose three future research directions: resilience for control, resource management of security mechanisms, and resilient CPS architecture.

1) Resilience for Control: By the cyber-physical attacks, the physical state diverges to infinity. However, most cyber-attack detection methods focus on the detectability of the attacks without properly considering the required time for detection. Future attack detection studies must highlight real-time detection performance with time-to-detect metrics. Furthermore, resilient CPS must include not only attack detection, but also attacker isolation and recovery from the effects of the attack.

2) Resource Management of Security Mechanisms: In realistic scenarios, CPS can have some limitations such as network and computing resources, conventional system extension, and battery capacity. For example, machine-learning-based resilient CPS functions enhance the resiliency, but accelerate battery consumption in battery-powered CPS. Also, frequent sensor measurements enhance the security level, but may violate the delay constraints of the physical systems owing to network congestion. To implement resilient CPS designs in real systems, resilient CPS mechanisms must properly consider these CPS resource limitations.

3) Resilient Architecture Design: To enhance the CPS security level, cross-layer security should be considered. For example, DoS attacks are implemented in various layers of the OSI network model, which can be detected in the physics-based detectors. However, to fundamentally remove DoS attacks, intrusion detection systems and attack handling methods are required at the network layer. Furthermore, the cross-layer security can reveal stealthy cyber-physical attacks. A physical stealthy attack cannot be detected by physics-based detectors. However, this stealthy attack can be easily detected by detecting abnormal behaviors on the networks. In contrast, a cyber-physical attack with unknown network vulnerabilities can be detected by physics-based anomaly detectors with the system dynamics.

H. Discussion on the Integration of Control Theory

As an integration of networks and control systems, cyber-physical security extends a dimension of the physical state in conventional IT security. While network-only security methods do not reflect the physical system disruption, considering the physical state provides control failure under cyber-physical attacks. Physical systems have operation ranges of the physical state, where the violation of the operation ranges indicates an irreparable state of the physical system.

We can derive the deadline constraints for CPS under cyber-physical attacks from mathematical system models and the operation range. The deadline constraint is essential for designing attack-resilient CPS; CPS function for resiliency must be executed before the deadline to prevent the physical system disruption. The execution of the function ensures the stability of the physical system before the physical state enters an irreparable state.

Furthermore, the conventional network-based security strategies may defend the cyber-physical attacks through the

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

networks. However, there are some limitations for network-based security to deal with cyber-physical attacks. The conventional network intrusion detection systems cannot reveal the cyber-physical attacks that exploit network vulnerabilities. Then, the control-theoretical attack detection algorithms can compensate make up for the vulnerabilities of the network vulnerabilities. When the sensor measurement or the control signal are manipulated, the physics-based anomaly detectors on the computing systems can detect the cyber-physical attacks when the cyber-physical attacks are launched by exploiting network vulnerabilities.

XI. Conclusion

Cyber-physical systems combined with wired and wireless networks can be compromised by malicious attackers through the networks. These attacks may cause significant damage to physical equipment and disastrous human casualties. To ensure the safety of CPS from cyber-physical attacks, it is necessary to analyze and detect cyber-physical attacks and to design CPS with reliability and resiliency.

This paper has provided a comprehensive analysis of cyber-physical attacks and has surveyed detection strategies against cyber-physical attacks from the perspective of networking. We have modeled CPS as hierarchical networked control systems, including the physical layer, the network layer, and the application layer. Then, we have introduced the functions of each layer. We have categorized cyber-physical attacks from three perspectives: the attack space, attack location, and stealthiness. Each of these three perspectives provides the requirements for cyber-physical attacks, i.e., the target data for manipulation, the intrusion channel, and the dynamics of physical systems. Moreover, we have discussed cyber-physical attack detection strategies in the context of physical dynamics, network anomalies, and machine learning techniques.

We have systematically analyzed the most common cyber-physical attacks, including the DoS attack, the replay attack, the zero-dynamics attack, the pole-dynamics attack, and the covert attack. We have further introduced the constraints on each attack implementation, the response of CPS to each attack with a numerical example, advanced attacks, and detection strategies for each attack. We have also discussed three resilient CPS design approaches: secure control, network management, and data-driven methods. Finally, we have identified possible future research directions with major research issues in resilient CPS design.

References

[1] 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.

[2] R. Rajkumar, I. Lee, L. Sha, and J. A. Stankovic, "Cyber-physical systems: The next computing revolution," in *Proc. Design Autom. Conf.*, 2010, pp. 731–736.

[3] 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.

[4] P. Park, S. C. Ergen, C. Fischione, C. Lu, and K. H. Johansson, "Wireless network design for control systems: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 2, pp. 978–1013, 2nd Quart., 2018.

[5] 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, Nov. 2018.

[6] K.-J. Park, H.-H. Lee, S. Choi, and K. Kang, "Design of a medical-grade QoS metric for wireless environments," *Trans. Emerg. Telecommun. Technol.*, vol. 27, no. 8, pp. 1022–1029, 2016.

[7] G. Aceto, V. Persico, and A. Pescapé, "A survey on information and communication technologies for industry 4.0: State-of-the-art, taxonomies, perspectives, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3467–3501, 4th Quart., 2019.

[8] J. Farooq and J. Soler, "Radio communication for communications-based train control (CBTC): A tutorial and survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1377–1402, 3rd Quart., 2017.

[9] B.-M. Cho, M.-S. Jang, and K.-J. Park, "Channel-aware congestion control in vehicular cyber-physical systems," *IEEE Access*, vol. 8, pp. 73193–73203, 2020.

[10] S. Son, K.-J. Park, and E.-C. Park, "Medical-grade channel access and admission control in 802.11e EDCA for healthcare applications," *PLoS ONE*, vol. 11, no. 8, 2016, Art. no. e0160052.

[11] D. Jin *et al.*, "Toward a cyber resilient and secure microgrid using software-defined networking," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2494–2504, Sep. 2017.

[12] A. Teixeira, D. Pérez, H. Sandberg, and K. H. Johansson, "Attack models and scenarios for networked control systems," in *Proc. Int. Conf. High Confidence Netw. Syst.*, 2012, pp. 55–64.

[13] I. Makhdoom, M. Abolhasan, J. Lipman, R. P. Liu, and W. Ni, "Anatomy of threats to the Internet of Things," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1636–1675, 2nd Quart., 2019.

[14] T. Alladi, V. Chamola, and S. Zeadally, "Industrial control systems: Cyberattack trends and countermeasures," *Comput. Commun.*, vol. 155, pp. 1–8, Apr. 2020.

[15] M. Pajic, J. Weimer, N. Bezzo, O. Sokolsky, G. J. Pappas, and I. Lee, "Design and implementation of attack-resilient cyberphysical systems: With a focus on attack-resilient state estimators," *IEEE Control Syst. Mag.*, vol. 37, no. 2, pp. 66–81, Apr. 2017.

[16] 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.

[17] M. Conti, N. Dragoni, and V. Lesyk, "A survey of man in the middle attacks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2027–2051, 3rd Quart., 2016.

[18] A. Cetinkaya, H. Ishii, and T. Hayakawa, "An overview on denial-of-service attacks in control systems: Attack models and security analyses," *Entropy*, vol. 21, no. 2, p. 210, 2019.

[19] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *Proc. Allerton Conf. Commun. Control Comput. (Allerton)*, 2009, pp. 911–918.

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

[21] 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.

[22] 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.

[23] J. Giraldo *et al.*, "A survey of physics-based attack detection in cyber-physical systems," *ACM Comput. Surveys*, vol. 51, no. 4, pp. 1–36, 2018.

[24] W. Zeng and M.-Y. Chow, "A trade-off model for performance and security in secured networked control systems," in *Proc. IEEE Int. Symp. Ind. Electron.*, 2011, pp. 1997–2002.

[25] X. D. Koutsoukos, "Systems science of secure and resilient cyberphysical systems," *Computer*, vol. 53, no. 3, pp. 57–61, 2020.

[26] D. Gunatilaka and C. Lu, "REACT: An agile control plane for industrial wireless sensor-actuator networks," in *Proc. IEEE/ACM Int. Conf. Internet Things Design Implement. (IoTDI)*, 2020, pp. 53–65.

[27] S. M. Dibaji, M. Pirani, D. B. Flamholz, A. M. Annaswamy, K. H. Johansson, and A. Chakrabortty, "A systems and control perspective of CPS security," *Annu. Rev. Control*, vol. 47, pp. 394–411, May 2019. [Online]. Available: https://doi.org/10.1016/j.arcontrol.2019.04.011

[28] 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. 44219–44227, 2020.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

[29] M. S. Chong, H. Sandberg, and A. M. H. Teixeira, "A tutorial introduction to security and privacy for cyber-physical systems," in *Proc. Eur. Control Conf. (ECC)*, 2019, pp. 968–978.

[30] M. Babaei, J. Shi, and S. Abdelwahed, "A survey on fault detection, isolation, and reconfiguration methods in electric ship power systems," *IEEE Access*, vol. 6, pp. 9430–9441, 2018.

[31] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, "IoT: Internet of Threats? A survey of practical security vulnerabilities in real IoT devices," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8182–8201, Oct. 2019.

[32] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, "Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on Internet-scale IoT exploitations," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2702–2733, 3rd Quart., 2019.

[33] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security—A survey," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1802–1831, Dec. 2017.

[34] A. K. Sikder, G. Petracca, H. Aksu, T. Jaeger, and A. S. Uluagac, "A survey on sensor-based threats and attacks to smart devices and applications," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 1125–1159, 2nd Quart., 2021.

[35] A. Alwarafy, K. A. Al-Thelaya, M. Abdallah, J. Schneider, and M. Hamdi, "A survey on security and privacy issues in edge-computing-assisted Internet of Things," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4004–4022, Mar. 2021.

[36] A. Isidori, *Nonlinear Control Systems*, 3rd ed. London, U.K.: Springer, 1995.

[37] H. Wang, H. Zhao, J. Zhang, D. Ma, J. Li, and J. Wei, "Survey on unmanned aerial vehicle networks: A cyber physical system perspective," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1027–1070, 2nd Quart., 2020.

[38] Y.-M. Kwon, J. Yu, B.-M. Cho, Y. Eun, and K.-J. Park, "Empirical analysis of MAVLink protocol vulnerability for attacking unmanned aerial vehicles," *IEEE Access*, vol. 6, pp. 43203–43212, 2018.

[39] C. Wang, C. Gill, and C. Lu, "FRAME: Fault tolerant and real-time messaging for edge computing," in *Proc. IEEE Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2019, pp. 976–985.

[40] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an international standard for wireless access in vehicular environments," in *Proc. VTC Spring IEEE Veh. Technol. Conf.*, 2008, pp. 2036–2040.

[41] 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.

[42] B. Dutertre, "Formal modeling and analysis of the Modbus protocol," in *Proc. Int. Conf. Crit. Infrast. Protection*, 2007, pp. 189–204.

[43] A. Kleinman and A. Wool, "Accurate modeling of the Siemens S7 SCADA protocol for intrusion detection and digital forensics," *J. Digit. Forensics Security Law*, vol. 9, no. 2, pp. 37–50, 2014.

[44] J. Song *et al.*, "WirelessHART: Applying wireless technology in real-time industrial process control," in *Proc. IEEE Real Time Embedded Technol. Appl. Symp.*, 2008, pp. 377–386.

[45] D. Gislason, *ZigBee Wireless Networking*. Burlington, MA, USA: Newnes, 2008.

[46] D. Gunatilaka and C. Lu, "Conservative channel reuse in real-time industrial wireless sensor-actuator networks," in *Proc. IEEE Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2018, pp. 344–353.

[47] *IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3)*, IEEE Standard 1815-2012, 2014.

[48] N. Higgins, V. Vyatkin, N. C. Nair, and K. Schwarz, "Distributed power system automation with IEC 61850, IEC 61499, and intelligent control," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 41, no. 1, pp. 81–92, Jan. 2011.

[49] S. H. Bouk, S. H. Ahmed, K.-J. Park, and Y. Eun, "Efficient data broadcast mitigation in multisource named-content discovery for vehicular CPS," *IEEE Commun. Lett.*, vol. 23, no. 9, pp. 1644–1647, Sep. 2019.

[50] C.-T. Chen, *Linear System Theory and Design*, 3rd ed. Oxford, U.K.: Oxford Univ. Press, 1998.

[51] D. S. Lakew, U. Sa'ad, N. Dao, W. Na, and S. Cho, "Routing in flying ad hoc networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1071–1120, 2nd Quart., 2020.

[52] 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.

[53] C. Schellenberger and P. Zhang, "Detection of covert attacks on cyber-physical systems by extending the system dynamics with an auxiliary system," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2017, pp. 1374–1379.

[54] A. K. Sikder, H. Aksu, and A. S. Uluagac, "6thSense: A context-aware sensor-based attack detector for smart devices," in *Proc. 26th USENIX Security Symp. (USENIX Security)*, 2017, pp. 397–414.

[55] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using Kalman filter," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 4, pp. 370–379, Dec. 2014.

[56] D. B. Rawat and C. Bajracharya, "Detection of false data injection attacks in smart grid communication systems," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1652–1656, Feb. 2015.

[57] S. Kang, S. Sristi, J. Karachiwala, and Y.-C. Hu, "Detection of anomaly in train speed for intelligent railway systems," in *Proc. Int. Conf. Control Autom. Diagnosis (ICCAD)*, 2018, pp. 1–6.

[58] W. Yan, L. K. Mestha, and M. Abbaszadeh, "Attack detection for securing cyber physical systems," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8471–8481, Oct. 2019.

[59] M.-K. Yoon, B. Liu, N. Hovakimyan, and L. Sha, "VirtualDrone: Virtual sensing, actuation, and communication for attack-resilient unmanned aerial systems," in *Proc. ACM/IEEE Int. Conf. Cyber Phys. Syst. (ICCPS)*, 2017, pp. 143–154.

[60] D. I. Urbina, J. A. Giraldo, N. O. Tippenhauer, and A. A. Cárdenas, "Attacking Fieldbus communications in ICS: Applications to the SWaT testbed," in *Proc. Singapore Cyber Security R D Conf.*, 2016, pp. 75–89.

[61] H. Choi *et al.*, "Detecting attacks against robotic vehicles: A control invariant approach," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, 2018, pp. 801–816.

[62] R. Quinonez, J. Giraldo, L. Salazar, E. Bauman, A. Cardenas, and Z. Lin, "SAVIOR: Securing autonomous vehicles with robust physical invariants," in *Proc. USENIX Security Symp. (USENIX Security)*, 2020, pp. 895–912.

[63] H. R. Ghaeini, D. Antonioli, F. Brasser, A.-R. Sadeghi, and N. O. Tippenhauer, "State-aware anomaly detection for industrial control systems," in *Proc. ACM Symp. Appl. Comput.*, 2018, pp. 1620–1628.

[64] W. Zhang, B. Bu, and H. Wang, "An intrusion detection method of data tampering attack in communication-based train control system," in *Proc. IEEE Intell. Transp. Syst. Conf.*, 2019, pp. 345–350.

[65] B. P. Poudel, A. Mustafa, A. Bidram, and H. Modares, "Detection and mitigation of cyber-threats in the DC microgrid distributed control system," *Int. J. Elect. Power Energy Syst.*, vol. 120, Sep. 2020, Art. no. 105968.

[66] A. Mustafa, B. Poudel, A. Bidram, and H. Modares, "Detection and mitigation of data manipulation attacks in AC microgrids," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2588–2603, May 2020.

[67] G. Fagogenis, V. De Carolis, and D. M. Lane, "Online fault detection and model adaptation for underwater vehicles in the case of thruster failures," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2016, pp. 2625–2630.

[68] 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. 2018.

[69] F. van Wyk, Y. Wang, A. Khojandi, and N. Masoud, "Real-time sensor anomaly detection and identification in automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 1264–1276, Mar. 2020.

[70] J. Shin, Y. Baek, Y. Eun, and S. H. Son, "Intelligent sensor attack detection and identification for automotive cyber-physical systems," in *Proc. IEEE Symp. Comput. Intell. (SSCI)*, 2017, pp. 1–8.

[71] H. Karimipour, A. Dehghantanha, R. M. Parizi, K.-K. R. Choo, and H. Leung, "A deep and scalable unsupervised machine learning system for cyber-attack detection in large-scale smart grids," *IEEE Access*, vol. 7, pp. 80778–80788, 2019.

[72] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1644–1652, Sep. 2017.

[73] S. A. Salinas and P. Li, "Privacy-preserving energy theft detection in microgrids: A state estimation approach," *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 883–894, Mar. 2016.

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

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

[75] D. Umsonst, H. Sandberg, and A. A. Cárdenas, "Security analysis of control system anomaly detectors," in *Proc. Amer. Control Conf. (ACC)*, 2017, pp. 5500–5506.

[76] B. Brumback and M. Srinath, "A chi-square test for fault-detection in Kalman filters," *IEEE Trans. Autom. Control*, vol. AC-32, no. 6, pp. 552–554, Jun. 1987.

[77] Y. Mo, R. Chabukswar, and B. Sinopoli, "Detecting integrity attacks on SCADA systems," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 4, pp. 1396–1407, Jul. 2014.

[78] Y. Shoukry, P. Martin, Y. Yona, S. Diggavi, and M. Srivastava, "Pycra: Physical challenge-response authentication for active sensors under spoofing attacks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, 2015, pp. 1004–1015.

[79] Y. Li and T. Chen, "Stochastic detector against linear deception attacks on remote state estimation," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2016, pp. 6291–6296.

[80] J. Huang, D. W. Ho, F. Li, W. Yang, and Y. Tang, "Secure remote state estimation against linear man-in-the-middle attacks using watermarking," *Automatica*, vol. 121, Nov. 2020, Art. no. 109182.

[81] S. Weerakkody, B. Sinopoli, S. Kar, and A. Datta, "Information flow for security in control systems," in *Proc. IEEE 55th Conf. Decis. Control (CDC)*, 2016, pp. 5065–5072.

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

[83] A. K. Sikder, H. Aksu, and A. S. Uluagac, "A context-aware framework for detecting sensor-based threats on smart devices," *IEEE Trans. Mobile Comput.*, vol. 19, no. 2, pp. 245–261, Feb. 2020.

[84] P. M. Van Every, M. Rodriguez, C. B. Jones, A. A. Mammoli, and M. Martínez-Ramón, "Advanced detection of HVAC faults using unsupervised SVM novelty detection and Gaussian process models," *Energy Build.*, vol. 149, pp. 216–224, Aug. 2017.

[85] M. Wilhelm, I. Martinovic, J. B. Schmitt, and V. Lenders, "Short paper: Reactive jamming in wireless networks: How realistic is the threat?" in *Proc. ACM Conf. Wireless Netw. Security*, 2011, pp. 47–52.

[86] H. Zhang, P. Cheng, L. Shi, and J. Chen, "Optimal DoS attack scheduling in wireless networked control system," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 3, pp. 843–852, May 2016.

[87] S. Lakshminarayana *et al.*, "Signal jamming attacks against communication-based train control: Attack impact and countermeasure," in *Proc. ACM Conf. Security Privacy Wireless Mobile Netw.*, 2018, pp. 160–171.

[88] S.-Y. Chang, B. A. N. Tran, Y.-C. Hu, and D. L. Jones, "Jamming with power boost: Leaky waveguide vulnerability in train systems," in *Proc. IEEE Int. Conf. Parallel Distrib. Syst. (ICPADS)*, 2015, pp. 37–43.

[89] W. Kim, J. Park, J. Jo, and H. Lim, "Covert jamming using fake ACK frame injection on IEEE 802.11 wireless LANs," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1502–1505, Oct. 2019.

[90] A. Haydari and Y. Yilmaz, "Real-time detection and mitigation of DDoS attacks in intelligent transportation systems," in *Proc. Int. Conf. Intell. Transp. Syst. (ITSC)*, 2018, pp. 157–163.

[91] S. H. Rose and T. Jayasree, "Detection of jamming attack using timestamp for WSN," *Ad Hoc Netw.*, vol. 91, Aug. 2019, Art. no. 101874.

[92] A. A. Fadele, M. Othman, I. A. T. Hashem, I. Yaqoob, M. Imran, and M. Shoaib, "A novel countermeasure technique for reactive jamming attack in Internet of Things," *Multimedia Tools Appl.*, vol. 78, no. 21, pp. 29899–29920, 2019.

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

[94] C. Fang, Y. Qi, P. Cheng, and W. X. Zheng, "Optimal periodic watermarking schedule for replay attack detection in cyber–physical systems," *Automatica*, vol. 112, Feb. 2020, Art. no. 108698.

[95] J. Kim, G. Park, H. Shim, and Y. Eun, "Zero-stealthy attack for sampled-data control systems: The case of faster actuation than sensing," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2016, pp. 5956–5961.

[96] J. Kim, J. Back, G. Park, C. Lee, H. Shim, and P. G. Voulgaris, "Neutralizing zero dynamics attack on sampled-data systems via generalized holds," *Automatica*, vol. 113, Mar. 2020, Art. no. 108778.

[97] A. Hoehn and P. Zhang, "Detection of covert attacks and zero dynamics attacks in cyber-physical systems," in *Proc. Amer. Control Conf. (ACC)*, 2016, pp. 302–307.

[98] 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. ACM/IEEE Int. Conf. Cyber Phys. Syst. (ICCPS)*, 2019, pp. 141–150.

[99] H. Jafarnejadsani, H. Lee, N. Hovakimyan, and P. Voulgaris, "Dual-rate L1 adaptive controller for cyber-physical sampled-data systems," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2017, pp. 6259–6264.

[100] S. Kim, Y. Eun, and K.-J. Park, "Stealthy sensor attack detection and real-time performance recovery for resilient CPS," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7412–7422, Nov. 2021.

[101] Z. Li and G.-H. Yang, "A data-driven covert attack strategy in the closed-loop cyber-physical systems," *J. Franklin Inst.*, vol. 355, no. 14, pp. 6454–6468, 2018.

[102] A. Praseed and P. S. Thilagam, "DDoS attacks at the application layer: Challenges and research perspectives for safeguarding Web applications," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 661–685, 1st Quart., 2019.

[103] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems," *ACM Comput. Surveys*, vol. 39, no. 1, p. 3, 2007.

[104] A. Cetinkaya, H. Ishii, and T. Hayakawa, "A probabilistic characterization of random and malicious communication failures in multi-hop networked control," *SIAM J. Control Optim.*, vol. 56, no. 5, pp. 3320–3350, 2018.

[105] M. B. G. Cloosterman, N. van de Wouw, W. P. M. H. Heemels, and H. Nijmeijer, "Stability of networked control systems with uncertain time-varying delays," *IEEE Trans. Autom. Control*, vol. 54, no. 7, pp. 1575–1580, Jul. 2009.

[106] A. Cetinkaya, H. Ishii, and T. Hayakawa, "Networked control under random and malicious packet losses," *IEEE Trans. Autom. Control*, vol. 62, no. 5, pp. 2434–2449, May 2017.

[107] S. Feng and P. Tesi, "Networked control systems under denial-of-service: Co-located vs. remote architectures," *Syst. Control Lett.*, vol. 108, pp. 40–47, Oct. 2017.

[108] D. Ding, Z. Wang, G. Wei, and F. E. Alsaadi, "Event-based security control for discrete-time stochastic systems," *IET Control Theory Appl.*, vol. 10, no. 15, pp. 1808–1815, 2016.

[109] Y.-C. Sun and G.-H. Yang, "Periodic event-triggered resilient control for cyber-physical systems under denial-of-service attacks," *J. Franklin Inst.*, vol. 355, no. 13, pp. 5613–5631, 2018.

[110] A. Cuenca, D. J. Antunes, A. Castillo, P. García, B. A. Khashooei, and W. Heemels, "Periodic event-triggered sampling and dual-rate control for a wireless networked control system with applications to UAVs," *IEEE Trans. Ind. Electron.*, vol. 66, no. 4, pp. 3157–3166, Apr. 2019.

[111] J. Kim *et al.*, "Encrypting controller using fully homomorphic encryption for security of cyber-physical systems," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 175–180, 2016.

[112] J. Kim and H. Shim, "Encrypted state estimation in networked control systems," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2019, pp. 7190–7195.

[113] J. H. Cheon *et al.*, "Toward a secure drone system: Flying with real-time homomorphic authenticated encryption," *IEEE Access*, vol. 6, pp. 24325–24339, 2018.

[114] H. Wang, L. Xu, and G. Gu, "FloodGuard: A DoS attack prevention extension in software-defined networks," in *Proc. IEEE/IFIP Int. Conf. Depend. Syst. Netw.*, 2015, pp. 239–250.

[115] G. Shang, P. Zhe, X. Bin, H. Aiqun, and R. Kui, "FloodDefender: Protecting data and control plane resources under SDN-aimed DoS attacks," in *Proc. IEEE INFOCOM IEEE Conf. Comput. Commun.*, 2017, pp. 1–9.

[116] D. He, S. Chan, X. Ni, and M. Guizani, "Software-defined-networking-enabled traffic anomaly detection and mitigation," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1890–1898, Dec. 2017.

[117] T. Ha, S. Yoon, A. C. Risdianto, J. Kim, and H. Lim, "Suspicious flow forwarding for multiple intrusion detection systems on software-defined networks," *IEEE Netw.*, vol. 30, no. 6, pp. 22–27, Nov./Dec. 2016.

[118] 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.

[119] C. Li, Z. Qin, E. Novak, and Q. Li, "Securing SDN infrastructure of IoT–fog networks from MitM attacks," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1156–1164, Oct. 2017.

[120] K. Lee *et al.*, "MC-SDN: Supporting mixed-criticality real-time communication using software-defined networking," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6325–6344, Aug. 2019.

[121] X. Li, D. Li, J. Wan, C. Liu, and M. Imran, "Adaptive transmission optimization in SDN-based Industrial Internet of Things with edge computing," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1351–1360, Jun. 2018.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

[122] S. Feng, A. Cetinkaya, H. Ishii, P. Tesi, and C. De Persis, "Networked control under DoS attacks: Trade-offs between resilience and data rate," *IEEE Trans. Autom. Control*, vol. 66, no. 1, pp. 460–467, Jan. 2021.

[123] A. Cetinkaya, K. Kikuchi, T. Hayakawa, and H. Ishii, "Randomized transmission protocols for protection against jamming attacks in multi-agent consensus," *Automatica*, vol. 117, Jul. 2020, Art. no. 108960.

[124] Y. Ma, C. Lu, and Y. Wang, "Efficient holistic control: Self-awareness across controllers and wireless networks," *ACM Trans. Cyber Phys. Syst.*, vol. 4, no. 4, pp. 1–27, 2020.

[125] P. Park, P. Di Marco, and K. H. Johansson, "Cross-layer optimization for industrial control applications using wireless sensor and actuator mesh networks," *IEEE Trans. Ind. Electron.*, vol. 64, no. 4, pp. 3250–3259, Apr. 2017.

[126] 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.

[127] A. Ferdowsi, U. Challita, W. Saad, and N. B. Mandayam, "Robust deep reinforcement learning for security and safety in autonomous vehicle systems," in *Proc. Int. Conf. Intell. Transp. Syst. (ITSC)*, 2018, pp. 307–312.

[128] F. Wei, Z. Wan, and H. He, "Cyber-attack recovery strategy for smart grid based on deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2476–2486, May 2020.

[129] A. Keliris, H. Salehghaffari, B. Cairl, P. Krishnamurthy, M. Maniatakos, and F. Khorrami, "Machine learning-based defense against process-aware attacks on industrial control systems," in *Proc. IEEE Int. Test Conf. (ITC)*, 2016, pp. 1–10.

[130] X. Lou, C. Tran, R. Tan, D. K. Yau, and Z. T. Kalbarczyk, "Assessing and mitigating impact of time delay attack: A case study for power grid frequency control," in *Proc. ACM/IEEE Int. Conf. Cyber Phys. Syst. (ICCPS)*, 2019, pp. 207–216.

[131] N. Abuzainab *et al.*, "QoS and jamming-aware wireless networking using deep reinforcement learning," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, 2019, pp. 610–615.

[132] R. F. Atallah, C. M. Assi, and M. J. Khabbaz, "Scheduling the operation of a connected vehicular network using deep reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1669–1682, May 2019.

[133] Q. Guo, Y. Li, Y. Song, D. Wang, and W. Chen, "Intelligent fault diagnosis method based on full 1-D convolutional generative adversarial network," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 2044–2053, Mar. 2020.

[134] K. Vatanparvar and M. A. A. Faruque, "Self-secured control with anomaly detection and recovery in automotive cyber-physical systems," in *Proc. Design Autom. Test Europe Conf. Exhibit. (DATE)*, 2019, pp. 788–793.

[135] W. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2012, pp. 3270–3285.

[136] A. V. Proskurnikov and M. Mazo, Jr., "Lyapunov design for event-triggered exponential stabilization," in *Proc. Int. Conf. Hybrid Syst. Comput. Control (CPS Week)*, 2018, pp. 111–119.

[137] J. Liu, E. Tian, X. Xie, and H. Lin, "Distributed event-triggered control for networked control systems with stochastic cyber-attacks," *J. Franklin Inst.*, vol. 356, no. 17, pp. 10260–10276, 2019.

[138] S. Feng and P. Tesi, "Resilient control under denial-of-service: Robust design," *Automatica*, vol. 79, pp. 42–51, May 2017.

[139] H. Chen and B. Yang, "A performance evaluation of CAN encryption," in *Proc. 1st IEEE Int. Conf. Trust Privacy Security Intell. Syst. Appl. (TPS-ISA)*, 2019, pp. 140–149.

[140] *Information Security—Lightweight Cryptography—Part 2: Block Ciphers*, ISO/IEC Standard 29192-2, 2019.

[141] A. Bogdanov *et al.*, "PRESENT: An ultra-lightweight block cipher," in *Proc. Int. Workshop Cryptograph. Hardw. Embedded Syst.*, 2007, pp. 450–466.

[142] T. Shirai, K. Shibutani, T. Akishita, S. Moriai, and T. Iwata, "The 128-bit blockcipher CLEFIA," in *Proc. Int. Workshop Fast Softw. Encrypt.*, 2007, pp. 181–195.

[143] D. Hong, J.-K. Lee, D.-C. Kim, D. Kwon, K. H. Ryu, and D.-G. Lee, "LEA: A 128-bit block cipher for fast encryption on common processors," in *Proc. Int. Workshop Inf. Security Appl.*, 2013, pp. 3–27.

[144] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, and L. Wingers, "The SIMON and SPECK families of lightweight block ciphers," IACR Cryptol. ePrint Arch., Lyon, France, Rep. 2013/404, 2013. [Online]. Available: https://eprint.iacr.org/2013/404

[145] M. M. Fouda, Z. M. Fadlullah, N. Kato, R. Lu, and X. S. Shen, "A lightweight message authentication scheme for smart grid communications," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 675–685, Dec. 2011.

[146] M. S. Turan *et al.*, *Status Report on the First Round of the NIST Lightweight Cryptography Standardization Process*, Nat. Inst. Stand. Technol., Gaithersburg, MD, USA, 2019.

[147] B. J. Mohd and T. Hayajneh, "Lightweight block ciphers for IoT: Energy optimization and survivability techniques," *IEEE Access*, vol. 6, pp. 35966–35978, 2018.

[148] F. Hu, Q. Hao, and K. Bao, "A survey on software-defined network and OpenFlow: From concept to implementation," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 2181–2206, 4th Quart., 2014.

[149] M. A. Togou, D. A. Chekired, L. Khoukhi, and G.-M. Muntean, "A distributed control plane for path computation scalability in software-defined networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2018, pp. 1–6.

[150] H. Yang, K. Zhan, M. Kadoch, Y. Liang, and M. Cheriet, "BLCS: Brain-like distributed control security in cyber physical systems," *IEEE Netw.*, vol. 34, no. 3, pp. 8–15, May/Jun. 2020.

[151] K. Sood, S. Yu, and Y. Xiang, "Software-defined wireless networking opportunities and challenges for Internet-of-Things: A review," *IEEE Internet Things J.*, vol. 3, no. 4, pp. 453–463, Aug. 2016.

[152] D. Kim, Y. Won, Y. Eun, and K.-J. Park, "W-simplex: Resilient network and control co-design under wireless channel uncertainty in cyber-physical systems," in *Proc. IEEE Conf. Control Technol. Appl. (CCTA)*, 2017, pp. 49–54.

[153] S. H. Bouk, S. H. Ahmed, K.-J. Park, and Y. Eun, "Interest broadcast suppression scheme for named data wireless sensor networks," *IEEE Access*, vol. 7, pp. 51799–51809, 2019.

[154] 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. ACM/IEEE Int. Conf. Cyber Phys. Syst. (ICCPS)*, 2019, pp. 87–96.

[155] D. Kim, Y. Won, Y. Eun, and K.-J. Park, "Resilient architecture for network and control co-design under wireless channel uncertainty in cyber-physical systems," *Trans. Emerg. Telecommun. Technol.*, vol. 30, no. 4, 2019, Art. no. e3499.

[156] X. Jin, A. Saifullah, C. Lu, and P. Zeng, "Real-time scheduling for event-triggered and time-triggered flows in industrial wireless sensor–actuator networks," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, 2019, pp. 1684–1692.

[157] Y. Ma *et al.*, "Optimal dynamic scheduling of wireless networked control systems," in *Proc. ACM/IEEE Int. Conf. Cyber Phys. Syst. (ICCPS)*, 2019, pp. 77–86.

[158] L. Xiao, Y. Li, G. Han, G. Liu, and W. Zhuang, "PHY-layer spoofing detection with reinforcement learning in wireless networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 10037–10047, Dec. 2016.

[159] Á. L. P. Gómez *et al.*, "On the generation of anomaly detection datasets in industrial control systems," *IEEE Access*, vol. 7, pp. 177460–177473, 2019.

[160] C. Yinka-Banjo and O.-A. Ugot, "A review of generative adversarial networks and its application in cybersecurity," *Artif. Intell. Rev.*, vol. 53, no. 1, pp. 1721–1736, 2020.

[161] S. Baluja and I. Fischer, "Learning to attack: Adversarial transformation networks," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 2687–2695.

[162] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S.-K. Ng, "MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks," in *Proc. Int. Conf. Artif. Neural Netw.*, 2019, pp. 703–716.

[163] W. Wu, S. Reimann, D. Görges, and S. Liu, "Event-triggered control for discrete-time linear systems subject to bounded disturbance," *Int. J. Robust Nonlinear Control*, vol. 26, no. 9, pp. 1902–1918, 2016.

[164] C. Wu *et al.*, "Maximizing network lifetime of WirelessHART networks under graph routing," in *Proc. IEEE Int. Conf. Internet Things Design Implement. (IoTDI)*, 2016, pp. 176–186.

[165] R. Brummet, D. Gunatilaka, D. Vyas, O. Chipara, and C. Lu, "A flexible retransmission policy for industrial wireless sensor actuator networks," in *Proc. IEEE Int. Conf. Ind. Internet (ICII)*, 2018, pp. 79–88.

[166] C. Wang, C. Gill, and C. Lu, "Adaptive data replication in real-time reliable edge computing for Internet of Things," in *Proc. IEEE/ACM Int. Conf. Internet Things Design Implement. (IoTDI)*, 2020, pp. 128–134.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.

Sangjun Kim received the B.S. degree in electronics engineering from Pukyong National University, Busan, South Korea, in 2017. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Computer Science, Daegu Gyeongbuk Institute of Science and Technology, Daegu, South Korea. His research interests include cyber–physical systems and software defined networking.

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 at Urbana–Champaign, Champaign, IL, USA. He is currently a Professor with the Department of Electrical Engineering and Computer Science, Daegu Gyeongbuk Institute of Science and Technology, Daegu, South Korea. His research interests include cyber–physical systems, industrial communications, and smart manufacturing.

Chenyang Lu (Fellow, IEEE) received the Ph.D. degree from the University of Virginia in 2001. He is the Fullgraf Professor with the Department of Computer Science and Engineering, Washington University in St. Louis. The author and co-author of over 200 research papers with over 24 000 citations and an H-index of 72. His research interests include embedded and real-time systems, cyber–physical systems, Internet of Things, mobile health, and clinical AI. He is the Editor-in-Chief of *ACM Transactions on Cyber-Physical Systems*. He also served as the Editor-in-Chief of *ACM Transactions on Sensor Networks* from 2011 to 2017 and the Chair of the IEEE Technical Committee on Real-Time Systems from 2018 to 2019. He is a Fellow of ACM.

Authorized licensed use limited to: DGIST. Downloaded on September 01,2022 at 08:42:08 UTC from IEEE Xplore. Restrictions apply.