IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 4, 2018, accepted September 3, 2018, date of publication September 13, 2018, date of current version October 12, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2869834

Review on Testing of Cyber Physical Systems: Methods and Testbeds

XIN ZHOU, (Member, IEEE), XIAODONG GOU, (Member, IEEE), TINGTING HUANG, (Member, IEEE), AND SHUNKUN YANG, (Member, IEEE)
School of Reliability and Systems Engineering, Beihang University, Beijing 100083, China

Corresponding author: Shunkun Yang (ysk@buaa.edu.cn)

This work was supported in part by the NSFC under Grant 61672080, in part by the JWYY under Grant 41402020502, and in part by the NASFC under Grant 2016ZD51031.

ABSTRACT Cyber physical systems (CPSs) are rapidly developing, with increasing scale, complexity, and heterogeneity. However, testing CPSs systematically to ensure that they operate with high reliability remains a big challenge. Therefore, it is necessary to summarize existing works and technologies systematically, with the aim of inspiring new inventions for more efficient CPS testing. Accordingly, this paper first investigated the advances in CPS testing methods from ten aspects, including different testing paradigms, technologies, and some non-functional testing methods (including security testing, robust testing, and fragility testing). Then, we further elaborate on the infrastructures of CPS testbeds from the perspectives of their architecture and the corresponding function analyses. Finally, challenges and future research directions are identified and discussed. It can be concluded that future CPS testing should focus more on the combination of different paradigms and technologies for multi-objective by integrating more emerging cutting-edge technologies such as Internet of things, big data, cloud computing, and AI.

INDEX TERMS Cyber-physical system, non-functional testing, robust testing, security testing, testing method, testbed.

I. INTRODUCTION

Cyber-Physical Systems (CPSs) are becoming more prevalent and their applications have penetrated the fields of aerospace, transportation, critical infrastructure, and industrial manufacturing [1], [2]. A CPS is typically a complex system with the following characteristics: data-driven, software definition, ubiquitous connection, mapping between virtual and reality, heterogeneous integration, and self-government [24]. With the increasingly extensive and in-depth applications of CPSs, the importance of their reliability and dependability cannot be overstated.

However, due to the large scale, high complexity, and heterogeneity of CPSs, their failure rate is high and there are many security risks, which can lead to huge economic losses or security breaches [26]. Reliability, safety, and privacy of CPSs are therefore significantly important [3], [4]. In order to realize the qualified operation of CPSs, the reliability of hardware, dependability of software, and security of communication should be ensured [5]. Moreover, recovery from failure should also be realized quickly. Therefore, CPS testing is necessary before real world implementation.

Due to the great significance of CPSs, research on CPS testing methods and key technologies continues to attract widespread attention from all around the world. CPS testing has been done in the fields of automotives [6], smart grids [7], industrial automation [8], health care [9], and robotics [10]. Though many reviews on CPSs already existed in the literature [11]–[22], there is little focus on surveying CPS testing methods [13], [14]. As a result, we investigate the latest research advances of CPS testing by considering test methods and testbeds. For the testing methods, we focus mainly on mainstream testing paradigms and the underlying enabling technologies, as well as some non-functional testing objectives such as robustness and security testing methods. For the studies on testbeds, we consider the highly complex and interdisciplinary nature of CPS testing; we concentrate on CPS architecture, communications infrastructure, and the corresponding functional analysis.

The main contributions of this paper are as follows: 1) First, we investigate the recent advances of testing methods for CPS from ten aspects, including different testing paradigms and objectives. 2) Secondly, we elaborate on the infrastructures of existing CPS testbeds and emphasize their architecture

2169-3536 © 2018 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

and function analysis. 3) Thirdly, we summarize the challenges in developing new testing methods and testbeds for complex CPSs along with possible solutions to inspire new CPS research directions. In particular, this survey will provide a wide range of guidance for industry engineers so they can identify the most suitable testing methods and testbeds, while achieving the lowest cost, the most convenience, and efficient schema for specific application.

The remainder of this paper is organized as follows. Section II describes the relevant background knowledge of CPSs. Section III comprehensively describes CPS testing methods. Section IV introduces the existing CPS testbed. In Section V we summarize the challenges faced by the existing test methods and testbeds, as well as the possible future research issues. Finally, Section VI presents a conclusion to our paper.

II. RELATED WORKS

There are several existing in-depth reviews on CPSs whose focuses are two-fold: CPS design and application, and testing and verification.

A. REVIEWS ON CPS DESIGN AND APPLICATION

The development of CPSs is discussed from perspectives of system model, information processing technology, and software design in [23]. The advancements in the development and applications of CPSs is surveyed in [20]. A review of CPSs in healthcare is presented in [22]. Four main research challenges (security, energy consumption, mobile dynamic environment, and system stability) in mobile CPSs are discussed in [129]. Intensive literature on the frontiers of CPS security is reviewed and key research challenges are identified in [17] and [19]. The integration of cloud computing with CPSs is reviewed in [18]. The advances of research in the field of cloud-based services and big data analytics for smart manufacturing are reviewed in [16].

B. REVIEW ON CPS TESTING AND VERIFICATION

In [14], testing methods for CPSs are surveyed. For simulation-based testing and verification for embedded control systems, an overview of existing and emerging advanced techniques is given in [124]. In terms of CPS monitoring, the state-of-the-art theories and techniques for qualitative and quantitative specification-based monitoring of CPS behaviors are summarized in [125]. For testbeds, a comprehensive survey on smart grid CPSs is presented in [13], which focuses on smart grid domains, research goals, test platforms, and communication infrastructure.

Table 1 shows that most of the current CPS reviews and surveys are focused on CPS design and the most considered non-functional attribute of CPSs is security, while the reviews on CPS testing are relatively scarce, especially for non-functional CPS attributes other than security.

TABLE 1. Main focus of current CPS reviews & surveys.

Reference	Testing/V&V	Design	Non-functional attribute
[11],2014	-	-	-
[12],2016	-	√	-
[13],2017	testbed	√	-
[14],2015	methods		-
[15],2016	-	√	security
[16],2016	-	√	security
[17],2012	-	√	security
[18],2016	-	√	-
[19],2017	-	√	security
[20],2015	V&V	√	security/resilience
[23],2017	testbed	√	security
[129],2018	-	√	security

III. RECENT ADVANCES OF TESTING METHODS FOR CPSs

In this section we will introduce state-of-the-art CPS testing methods. As shown in Table 2, the existing works mainly focus on model-based testing, search-based testing, online monitor-based testing, fault injection-based testing, big data driven testing, cloud-based testing, and especially non-functional testing of security, conformance, and robustness.

TABLE 2. Summary of the research on CPS testing methods.

CPS Testing Methods		Year of publication	Reference
model-based	health care model	2012-2014	[29][30]
	robot model	2013-2015	[31][32]
	industrial automatic control system model	2015	[33]
	smart grid model	2016	[34]
	hybrid system model	2015	[35]
	general object model	2015-2016	[36][37][38]
search-based	genetic algorithms	2015-2016	[36][39]
	non-genetic algorithms	2013-2016	[40][41][42]
monitor-based	runtime verification	2012-2016	[43][44][45][46]
fault injection-based	reliability testing	2010-2016	[47][48][49]
big data driven	big data processing method	2014-2016	[50][51][52][53]
	big data analysis method	2015-2016	[54][55][16]
	CPS and big data fusion	2014-2016	[56][57][58][59]
cloud-based	health care cloud	2014-2015	[60][61]
	vehicle cloud	2002-2012	[62][63][64]
	security cloud	2011-2016	[65][66][67]
	crossing field cloud	2014-2017	[68][69][70][71][72]
others	conformance testing	2011-2016	[73][74][75][76][77][78]
	robustness testing	2010-2014	[79][80][81]
	security testing	2013-2016	[82][83][84][85][86][87][88]
	fragility testing	2017	[89]

A. TESTING METHODS CATEGORIZED BY THEIR PARADIGM AND UNDERLYING TECHNIQUES

1) MODEL-BASED CPS TESTING

Model-based testing (MBT) is a paradigm widely used in the field of software testing; it checks the correctness according to the expected behavior specified by the formalization

TABLE 3. Taxonomy of MBT for CPS.

MBT for CPS	Model	OL/CL	Delayed/Real-Time (DL/RT)	On-Line/Off-Line (ONL/OFL)	Active/Passive	Reference
Medical Cyber-Physical Systems	Ptolemy II	CL	RT	ONL	Active	[29]
U-test	UML	OL	DL	OFL	Passive	[36]
CPN based testing	CPN&FSM	OL	DL	OFL	Passive	[32]
NCES	NCES	CL	DL	OFL	Active	[33]
VHM	Automata	CL	RT	ONL	Active	[30]
SES for fidelity evaluation	Simulink	CL	RT	ONL	Active	[37]
DER	Neural Network	OL	RT	ONL	Active	[34]
Acumen for hybrid system model	Matlab& Automata	OL	RT	OFL	Passive	[35]
Security Pattern Evaluation	UML& MARTE	OL	RT	ONL	Active	[38]
State-based security evaluation	Game-theoretic	OL	DL	OFL	Passive	[39]

or model [27], [28]. In recent years, more attention has been paid to MBT of CPSs, specifically in the fields of health care [29], [30], robotics [31], [32], industrial automatic control systems [33], smart grids [34], hybrid systems [35], and the other general CPS objectives [36]–[38]. As shown in Table 3, CPSs can be modeled by different tools, including Simulink, UML, FSM, and Automata. The closed-loop (CL) mechanism has an feedback link as a control signal, which reacts to the physical components for improving the accuracy of the system. On the other hand, the open-loop (OL) testing mechanism does not consider feedback. Real-time MBT has strict timing constraints and deterministic behavior. Online MBT means that the MBT tools are directly connected to the system for dynamic testing in the context of a real world environment. The active testing focuses on the test case in which the entire test process is controlled by the test program, while the passive test means that the test process only involves monitoring. CPS models have been developed successfully for many different domains. Silva *et al.* [29] design a medical device model and patient model based on the Ptolemy II (V8.0.1) framework, which adopts an actor-oriented design (AOD) for the extraction of the clinical data of patient model information from the MIMIC II clinical database (V2.6); the model and data form a closed-loop test environment. Jiang *et al.* [30] complete a similar work, constructing a patient model that combs a real-time virtual heart model (VHM) with an integrated framework of implantation device validation. The heart model performs closed-loop equipment verification by presenting a synthetic electrogram signal to the device and responding to a functional pacing signal from the device. The interactive and physiologically relevant model-based test generation of pacemaker device operation constitutes a closed-loop test environment.

Zander [31] and Saglietti *et al.* [32] use Simulink to achieve a simulation cycle and simulation state retrieval during the implementation of a system model. They track the approximate error to improve performance and reliability of the system and evaluate the efficiency level of design execution. Some scholars extend colored petri nets (CPN) theory to CPS testing. Saglietti *et al.* [32] realize the automatic generation of test cases based on CPN coverage standard using a finite state machine (FSM) automatic analysis algorithm, and they found that the best performance algorithm is the so-called hot priority algorithm. Buzhinsky *et al.* [33] setup a new test framework through the modular formal language NCES from the perspective of formal verification technology. They represent the formal model of CPS testing in closed-loop form and use the model check method to verify the generated test cases. Based on the nonlinear artificial neural network model, Kosek and Gehrke [34] design an anomaly detection method to efficiently detect and evaluate the exception in Cyber-Physical Intrusion Detection in Smart Grids. Through verifying the measurement data, the accuracy of anomaly detection can be improved. A specific modeling language (Acumen) for the hybrid system model, which can implement MBT for CPSs in Matlab, is proposed by Aerts *et al.* [35]. Ali and Yue [36] plan to solve the problem of CPS uncertainty testing based on MBT. Schmidt *et al.* [37] apply the system entity structure (SES) on the abstract level to realize automatic high fidelity assessment of a complex and modular simulation model. Motii *et al.* [38] propose a method of evaluating security solutions based on UML modeling and extensively use modeling and analysis of real time and embedded systems (MARTE) to implement an architectural assessment of timing issues. Orojloo and Azgomi [39] propose a game-theoretic modeling approach to evaluate the security of CPSs, where a state-based model is structured and parameterized using a game theory approach to investigate the dynamic behavior of CPSs under attacks; then, the parameterized state-based model is transferred into a solvable model to quantify the security measures suitable for CPSs.

These different CPS models can be very useful as a basis for MBT of CPS, especially when there is no real CPS available for testing or some parts of the real CPS are not suitable for special abnormal and destructive testing due to safety or economic issues.

2) SEARCH-BASED CPS TESTING

Search-based test methods use meta-heuristic search techniques, such as genetic algorithm (GA), simulated annealing algorithm, and tabu search algorithm to automatically generate test data or test cases. In recent years, search-based

TABLE 4. Online monitor tools for CPS.

References	Featured Functions	Specification	Monitor	Subjects	Enable techniques
[128]	Monitor multiple specifications simultaneously	Manually	re-implemented JavaMOP	Software (java)	JavaMOP
[130]	adapting software model checker for runtime verification	automatically re-construct a description of the execution environment	Modifications to the model checker are small and self-contained	software	Model check; Software model checker
[138]	Runtime monitoring for error recovery	parametric properties with differed formalisms	generated monitoring code	software (Java)	AspectJ
[131]	static analysis based predicative runtime verification framework for CPSs	parameterizes static analysis methods	generate predicative words at compiling time	CPS、software	JavaMOP; parameterizes the static analysis methods
[127]	a method to synthesize monitors for CPS by theorem proving	Manually	automatically	CPS(water tank; auto cars; robots, etc.)	Differential dynamic logic; theorem proving
[132]	robust online monitoring of partial signals	Manually	Manually	CPS	Signal Temporal Logic; Robust Interval Semantics
[129]	a specification framework for runtime monitoring.	Manually	automatically	CPS	timed automata; Metric Temporal Logics
[46]	Runtime verification ensuring performance qualities for CPS.	Manually	automatically	CPS(rover)	Linear programing model
[133]	monitor the safety and security of robots	Manually	automatically	robotic (ROS)	-
[134]	systematically testing via model checking and STL based online monitoring system	Manually	Manually embedded	autonomous drone	Model check; Signal temporal logic (STL)
[135]	composing multiple runtime enforcers for CPS	Manually	Manually Embedded	Parrot minidrones	SMT solver
[136][137]	A tool for temporal logic falsification for hybrid systems	Manually	Matlab toolbox	Hybrid systems automotive	Stochastic optimization; Metric Temporal Logic

algorithms have also begun to be applied in the field of CPS testing. Ali and Yue [36] apply a search algorithm based on GA to a solution to CPS uncertainty testing. Arrieta *et al.* [40] use a GA-based algorithm to select the suitable test case for different test levels; compared with the alternating variable method (AVM) and greedy algorithm, this method can reduce the cost of CPS product line configuration testing by approximately 80%. Arrieta *et al.* [41] also propose a weight-based search algorithm for the configurability and variability of large CPSs (such as vehicle and aerospace) to determine the prioritized execution order of the test case, find the best solution, and reduce the time to detect failures. Matinnejad *et al.* [42] designed the search-based testing method for a continuous controller through model-in-loop (MIL), software-in-loop (SIL), and hardware-in-loop (HIL) for the automotive embedded system, covering different searching strategies, including random search, adaptive random search, hill climbing algorithm, and simulated annealing algorithm. Yue *et al.* [43] propose a semi-automatic and interactive configuration solution for CPSs based on the search algorithm.

3) ONLINE MONITORING FOR CPS TESTING

Complete testing or formal verification is impossible for complex CPSs, but as a lightweight verification technology, online monitoring and runtime verification provides one useful formal technique for improving the reliability of real or simulated systems. CPS monitoring is the act of runtime verification by observing and evaluating the temporal behaviors of real systems or simulated models. Monitoring algorithms can qualitatively and/or quantitatively verify the satisfaction of a formula ϕ by a signal w (not the whole system), which is much easier than traditional model-checking. Due to relatively low computational complexity of the monitoring algorithms, as shown in Table 4, a large body of software tools (Temporal Rover, MaC, Java PathExplorer, LOLA, RV-Monitor, MonPoly, LTLFO2Mon, etc.) have been developed over the last two decades. Some publicly available software tools (AMT, Breach, S-Taliro and U-Check, etc.) for real-time reasoning with Boolean or multi-valued temporal logics can monitor both real valued and Boolean signals.

Real-time monitoring of CPSs has been applied for automotive systems and medical devices [125]. Kane *et al.* [44] designed an external runtime monitor to analyze vehicle log data during the testing process to detect whether a critical advanced attribute is violated. CHARON (a modeling language based on hybrid automata) and an event-oriented runtime monitoring framework are used in [45] to verify a CPS system against a temporal logic security specification. A runtime verification system called Brace [46] is designed to ensure that the errors injected in a CPS application can be efficiently detected with less runtime monitor overhead. ubiquitous monitoring for industrial CPSs is realized through relay-assisted wireless sensor networks in order to solve the distributed parameter estimation problem in [47]. In order to

express the temporal constraint for distributed CPSs more intuitively, a new timestamp temporal logic (TTL) is proposed in [48], which can automatically generate logic code and programs to monitor the expressed timing constraints. ModelPlex [127] provides correctness guarantees for CPS executions at runtime; it combines offline verification of CPS models with runtime validation of system executions for compliance with the model.

These advances in recent studies demonstrate that online monitoring and runtime verification can be complementary to most CPS testing techniques. However, there are still some issues to overcome in practice: Lack of specifications: It is difficult to obtain the suitable specifications for complex CPSs to be checked against [128].

High monitoring overhead: Online monitor and runtime verification can usually incur some overhead, especially when checking many properties simultaneously [128]. This challenge remains open for most resource constrained CPSs [129].

Difficulty in monitor synthesis: Because of the continuous physical plant, sensor inaccuracies, and actuator disturbance, full CPS model validation is rather elusive. The rewriting capabilities and flexibility of adaptable monitor generation is thus an interesting question for future work [127].

Traditional performance measures (statistical methods or transformations such as Fourier's for the purpose of classification assess signals and detect occurring patterns or noise removal) should be further integrated with those newly developed verification-inspired formalisms for capturing hybrid aspects of behaviors [125].

4) FAULT INJECTION FOR CPS TESTING

Fault injection, which is a key testing technology, is very efficient for verifying and evaluating a system by producing failure artificially and consciously in order to speed up the occurrence of an error or failure. It can also be effective in CPS testing.

Faza *et al.* [49] apply fault injection to fault localization for testing smart grids. Similarly, Frohlich *et al.* [50] also found that noninvasive and deterministic fault injection testing can provide reliable evidence for critical real-time systems. Alternatively, Vedder *et al.* [51] devise a combination of failure injection (FI) and property-based testing (PBT) to detect unusual safety violations by injecting failure randomly in the execution sequence. By attacking control parameters including sensor measurements and controller signals, Orojloo and Azgomi [52] have presented a new method to assess the direct and indirect impact of attacks on a CPS. Based on the sensitivity analysis, the priority of attacks is determined to help identify the most vulnerable components and control parameters in the system.

It can be found that to verity the reliability, security, and resilience of CPSs, fault injection is an efficient technology. Considering the complex running environment and constraints of a CPS, different kinds of fault injection tools should be adopted for more CPS testing and verification in the future.

5) BIG DATA-DRIVEN CPS TESTING

Big data related analysis techniques can benefit testing greatly by quickly processing and storing large amounts of testing data in a CPS. In this section, we survey the latest research results to analyze the coupling mechanism between big data and CPS and apply big data-based CPS testing.

Lee *et al.* [53] discuss a systematic architecture for applying CPS in manufacturing called 5C, which can automate and centralize data processing, health assessment, and prognostics. In order to provide data service specification and deal with large amounts of data safely in a short time, Zhang [54] proposed a big data-driven CPS based on architecture analysis and design language (AADL). Marini and Bianchini [55] designed a ''data as a service'' approach to deal with the big data environment. Don and Min [56] design a real-time dynamic data processing framework for decision making of MCPS in health care applications. Wang and Wang [57] find that big data from different machines, network sensors, and systems can be used for intelligent prediction and diagnosis to improve the quality and productivity of manufacturing. Lee *et al.* [58] provide a comprehensive solution for industrial information analysis systems through a CPS based on the internet of things (IoT) and various data analysis methods so that modules can be reconfigured and exchanged. Babiceanu and Seker [16] propose a framework for developing predictive manufacturing CPSs by integrating IoT and big data algorithm analysis. Lee *et al.* [59] introduces intelligent forecasting information tools for the fusion of CPSs and PHM algorithms in big data environments. Niggemann *et al.* [60] propose a new concept that applies CPSs and big data to automatically learn the necessary model. They also design a cognitive-based reference architecture, including task-specific human-machine interfaces (HMI) and a feedback/control mechanism, to address the possible challenges. Jara *et al.* [61] identified several challenges and opportunities for the integration of big data into CPSs and propose a hybrid method to build models and discover behaviors based on large amounts of data. Zhong and Zhang [62] discuss the challenges faced by combining CPSs with big data and illustrate the demands for dynamic continuous modeling, such as time and space modeling.

Through the above analyses, we find that there is a commonality between big data technologies and CPS testing, which can provide a starting point for the future development of CPS testing.

6) CLOUD TESTING FOR CPS

With the advances of cloud computing and IoT, cloud-based CPSs can extend their functionality to address computational and storage constraints. The integration of cloud and CPSs has begun to attract increasing attention. For example, Cheng [63] focus on private cloud infrastructure to run

medical CPS applications based on OpenStack. Zhang *et al.* [64] propose a Healthcare CPS by defining a unified standard for a data collection layer, data management layer, and data-oriented service layer for distributed storage and parallel computing. Abid *et al.* [65] propose a novel V-Cloud architecture, including a vehicular cyber-physical system, vehicle-to-vehicle network, and vehicle-to-infrastructure network layer.

For cloud testing of CPSs, Hahanov *et al.* [66] propose a CPS for intelligent cloud traffic control and testing. Burger [67] realized the cloud-based test method by interaction between sensors, actuators, and CPSs to test a vehicle system. Based on service oriented architectures (SOA), Puttonen *et al.* [68] designed a production system and a smart-mobility system, which can be used to test system security. The security of cloud cyber-physical systems is discussed in detail by Reddy [69]. Azab and Eltoweiss [70] introduce a resilient and secure defense cloud ''Cooperative Autonomous Resilient Defense''. Karnouskos *et al.* [71] propose a concept called ''Cloud of Things'', which involves the integration of a CPS and cloud, which a major trend in the design, implementation, and interaction of the CPS-oriented environment that will reshape future industry, especially in monitoring and management. Karnouskos *et al.* [72] propose a service architecture that covers the basic needs of monitoring, management, data processing, and integration by considering disruptive technologies and concepts for future industrial systems. Nakauchi *et al.* [73] present a virtual mobile cloud network (VMCN), an architecture for a scalable real-time CPS based on virtualization-capable network infrastructure, and highly distributed edge clouds. Alam and Saddik [74] present a digital twin architecture reference model for a cloud-based CPS called C2PS. The model helps in identifying various degrees of basic and hybrid computation-interaction modes in this paradigm. Shu *et al.* [75] propose a novel architecture of CCPS (CCPSA) to provide flexible services and applications.

From the above survey and analysis, we can find that the current cloud-based method has realized the initial fusion with CPSs, which demonstrates that the theory and method of using the cloud can further be developed and embodied for CPS testing in the future.

B. TESTING METHODS CATEGORIZED BY THEIR TESTING OBJECTIVES

1) CONFORMANCE TESTING FOR CPSs

Conformance testing is an important testing activity to verify that a product satisfies its specified standards. The conformance testing of a CPS refers to the degree of compliance between the implementation and the required standards. Conformance testing for different products, processes, and services has been widely applied. However, for CPS conformance testing, due to the complexities of the scale and diversity of standards, the current research is relatively scarce. Recent research can be classified as either CPS conformance testing generation or CPS conformance measurement.

CPS conformance testing generation: Li *et al.* [79] defined the conformance relationship of timed input - output conformance (TIOCOC) for timed constraint automata (TCA), and an automatic testing method is proposed to generate testing cases from TCA. Woehrle *et al.* [80] proposed two different conformance testing methods. The first is the method of segmented state space traversal to achieve scalability of quantitative conformance testing for CPSs. The second method is to use the physical measurement for conformance testing using timed testing of a physical quantity (TTQ) or online testing tools TRON [81].

CPS conformance measurement: In [76] and [77], a conformance test method is proposed by confining conformance testing as a logical attribute forgery problem, and the metric temporal logic (MTL) theory is adopted to calculate conformance. In [78], the concept of conformance testing is redefined and an algorithm for CPS conformance testing is designed through sampling the output of each discrete point to determine its correctness.

Because the standards involved in complex CPSs are diverse (e.g., a CPS may need to meet multiple standards, and the standards for different CPSs are more complex), universal and configurable CPS conformance testing is still a challenge for future research.

2) ROBUSTNESS TESTING FOR CPS

Robustness testing, which is also known as fault tolerance testing, is often used to test whether a system can automatically resume or ignore faults in the event of a failure. Robustness is a very important capability of CPSs, and the method of testing the robustness is a valuable research issue. Because robustness testing is usually destructive, current CPS robustness tests are mostly conducted at the model level (such as with Matlab, Simulink, etc.).

For example, Abbas *et al.* [82] apply the MTL theory for robustness testing of CPSs. The Monte Carlo simulation method is used to implement the random walk algorithm in the space of system input to find the paths and counter-examples of violations of robustness properties. The method can be applied to a Simulink / StateflowTM (S / S) model and is implemented in the Matlab toolbox S-TALIRO. Abbas *et al.* [83] also proposed a robust-guided temporal logic testing (RGTLT) method based on the S-TALIRO toolbox. This method uses the MTL theory to quantify the robustness of a stochastic CPS. The Markov chain Monte Carlo algorithm is used to solve the global minimization problem of robustness. Fainekos *et al.* [84] provided a framework for testing the correctness and robustness of a CPS simulation model. They create a linear symbol model for each execution time step in a given SimulinkTM model. The test input supplied by users is then re-run in the symbol model and the robustness is tested in the execution trace.

Robustness testing for real CPS systems may have an irreversible impact on the system itself, resulting in serious losses. However, if it is only carried out in a simulation environment, some real conditions cannot be reflected.

Therefore, performing a safe and effective CPS robustness test in a real application environment is an important challenge.

3) SECURITY TESTING FOR CPSs

CPSs are highly connected through different networks, which makes them vulnerable to ubiquitous cyber-attacks. Extensive studies on security testing are becoming more common, in order to reveal security-related flaws for different CPSs [29], [30], [34], [38].

Many researchers focus on security testing for systems against cyber-attacks. Pan *et al.* [85] proposed a specification-based intrusion detection framework by adopting a Bayesian network to graphically encode the causal relationship as rules of an intrusion detection system (IDS). Liu *et al.* [86] proposed a new cyber physical fusion method, called abnormal traffic index state estimation (ATSE), to detect cyber-attacks of smart grid, such as malicious data injection (MDI) and system intrusion. Specifically, the traffic characteristics of the cyber network are combined with the inherent physical laws of the power system into a unified model to reduce the impact of unreliable measurements on state estimation. The cyber physical monitoring system (CPMS) [87] is another testing framework for detecting cyber-attacks of a smart meter. Based on the information fusion of abnormality detection of cyber intrusion and traffic flows, the intrusion detection system Snort can be used to monitor the flow of all smart meters and use energy conservation to check the conformance of meter readings. Luo *et al.* [88] proposed a new system observability-based framework, which can not only effectively monitor the system, but can also solve the structural fragility of the system when faced with cyber-attacks.

One of the difficulties in security testing of CPSs is detecting unrecognizable attacks. Pasqualetti *et al.* [89] describe the undetectable and unrecognizable attacks that may exist in a CPS from the perspective of system theory and graph theory. A centralized monitor using the geometric control theory is designed, and a distributed detection monitor using distributed control technology and parallel computing technology is used to detect the malicious external attacks facing a CPS. Chen *et al.* [90] designed a dynamic attack detector using the side initial state information, which can detect all detectable attacks. Based on computational geometry, Gawand *et al.* [91] can detect cyber-attacks by analyzing the abnormal output of the CPS control system.

Although security testing has been given much attention, it is still not possible to avoid security breaches completely. Furthermore, combining security testing with robustness testing and reliability testing is an interesting research direction for the future.

4) FRAGILITY TESTING FOR CPSs

Small disturbances may cause a system to collapse or even lead to catastrophic consequences. The fragility of a CPS is an important attribute; it reflects the sensitivity of the system responding to external disturbances or changes. Fragility testing is therefore becoming a new-birth area for CPS testing. Ma *et al.* [92] proposed a new fragility test method based on model execution and reinforcement learning techniques, which can quickly reveal the defects of a self-healing cyber system (SH-CPS) with the presence of environmental uncertainties.

As a result of the abovementioned methods, we can see that many traditional testing methods (such as MBT) can be transferred to CPS testing. However, the non-functional testing of CPSs, including robustness, security, fragility, reliability, and resilience testing, is still a great challenge and has had few breakthroughs. We also did not find any specific research on reliability and resilience testing of CPSs in the literature. In terms of CPS development, reliability, fragility, and resilience testing are more important to reveal the dynamic evolution mechanism of CPSs effectively, and this is worthy of more attention in future research.

IV. RECENT ADVANCES IN TESTBEDs FOR CPSs

The operation of CPS testing is too difficult to achieve without effective platform support. In this section, we focus on the existing testbeds for CPSs from the viewpoints of both architecture and functions; this is inspired by Cintuglu *et al.* [13], who reviewed the existing smart grid testbed systematically.

As shown in Table 5, existing CPS testbeds cover many application domains, including smart grid, transportation systems, unmanned aerial vehicles, robotics, smart home equipment, water systems, and other specific CPSs. The majority of CPS testbeds are simulation-based or semi-simulated. The number of fully hardware-based testbeds is low, and these are usually applied to small CPSs, such as robot and unmanned aerial vehicle systems. Furthermore, distributed platforms are more popular than centralized ones. From the viewpoint of function analysis for current CPS testbeds, security testing-oriented and control testing-oriented testbeds are the majority. The communication architecture of each testbed is summarized in Table 6.

A. ARCHITECTURE OF CPS TESTBEDS

Based on enabling execution techniques, the architecture of testbeds can be divided into three categories according to the corresponding implementation enabling technique: hardware-based, simulation-based, or hybrid platform. Hardware-based testbeds are the physical components and cyber components are real or emulation devices; in other words, a hardware-based testbed is composed of real-world devices. This is more common in the testing of small CPSs, such as in intelligent home device testbeds [93], unmanned aerial vehicle system testbeds [96], [117], [122], robot testbeds [99]–[101], smart grid test platforms [112], general CPS [118], cloud manufacturing CPS [119] and traffic vehicle system testbed [114]. In this testbed, Zhou *et al.* [114] build a platform that composed of miniature vehicles to emulate the real traffic system. Simulation-based testbeds are composed of physical

TABLE 5. Taxonomy of existing cyber-physical system testbed.

Testbed	Year of Publication	Targeted objective	Application Domain	Platform Architecture	Distribution Architecture	Reference
IPv6-Based Testbed	2013	Dynamic Remote Control	Home Devices	Hardware	Distributed	[93]
Real-Time Cyber-Physical Co-Simulation Testbed	2015	Cyber Security	Smart Grid	Real-Time Simulator	Centralized	[94][95]
Testbed for UAV	2011	Cooperative Control of Swarms of UAV	UAV	Hardware	Centralized, Distributed	[96]
EPIC	2013	Cyber Security	Networked Critical Infrastructures	Hybrid	Distributed	[97][98]
UPBOT	2010	Security and Artificial Intelligence	Robots	Hardware	Distributed	[99][100]
Worcester Polytechnic Institute	2015	Control Algorithms	Assistive Robotics	Hardware	Distributed	[101]
RCPS testbed	2015	Hardware and Software Fault Tolerance, Software Reconfiguration, and System Stability	Distributed and Resilient CPS	Simulator	Centralized	[102]
FACIES Testbed	2015/2017	Security of Communication	Water System	Hybrid	Distributed	[103][104]
CPSTCS Testbed	2015	Cyber Security	Critical Infrastructures	Hybrid	Distributed	[105]
SURE	2016	Security and Resilience	Transportation Networks	Simulator	Centralized	[106]
PowerCyber Testbed	2013	Wide Area Situational Awareness, Cyber security	Smart Grid	Real-Time Simulator	Distributed	[107][108]
Turtlebot Robots Testbed	2016	Self-Adaptive Control	Autonomous Robots	Simulator	Centralized	[109]
WSU Testbed	2015	Cyber Security	Smart Grid	Hybrid	Distributed	[110]
Open Heterogeneous Wireless Testbeds	2013	Application Software and Supporting Firmware	CPS	Simulator	Distributed	[111]
Cloud Communication - Based Remote Access Testbeds	2016	Remote control, Cyber Security, Wide Area Situational Awareness	Smart Grid	Hardware	Distributed	[112]
Microgrids Testbed	2016	System Performance	Smart Grid	Real-Time Simulator	Distributed	[113]
Multi-Agent Testbed	2014	Cooperative Communication and Control	Vehicle Systems	Hardware	Distributed	[114]
Florida State University	2013	Distribution Grid Management, Demand Response	Smart Grid	Real-Time Simulator	Distributed	[115]
GISOO	2013	Interactions of Communication, Computation and Control Components	Wireless CPS	Simulator	Centralized	[116]
Up and Away	2014	Visually-Control	UAV	Hardware	Centralized	[117]
INVITED	2017	Timing Behavior	CPS	Hardware	Distributed	[118]
CPMC	2017	Monitor and Execute Manufacturing Operations	Cloud manufacturing CPS	Hardware	Distributed	[119]
Multifunctional CPS testbed	2017	Cyber Security	Smart grid	Simulator	Distributed	[120]
South Dakota State University	2017	Cyber Security and Stability Control	Smart grid	Simulator	Distributed	[121]
Drone-Be-Gone	2017	Autonomous Control	UAVs	Hardware	Centralized/ Distributed	[122]
WADI	2017	Cyber and Physical Attacks, Cascading Effects of Attacks	Water distribution system	Hardware	Centralized	[123]

components and cyber components and are implemented through simulation tools or programming. Most current CPS testbeds are simulation-based because the scale of some CPSs is too large or the cost of the physical hardware is too high; examples of simulation-based testbeds are distributed resilient CPS testbeds [102], traffic network testbeds [106], robot testbeds [109], testbeds for software applications and firmware of CPS [111], and testbeds for visual control of wireless CPS [116]. One further simulation-based testbed is a special testbed that uses real-time digital simulation technology (RTDS), named real-time simulator, such as [94], [95], [107], [108], [113] [115], [120], and [121]. The tools used for simulation-based testbeds are summarized in Table 7. Hybrid testbeds for CPSs are composed of real physical components and simulation network components, or simulated physical components and real network components; examples of these include testbeds for network critical infrastructure [97], [98], [105]. The testbed given in [97] and [98] is composed of a real network and simulated physics, and the testbed is composed of simulated network and real physics [105]. In addition, there is a testbed for water systems [103] and a testbed for smart grids [110]. In summary, simulation-based or hybrid testbeds are easy to implement and can guarantee safe execution of testing and repeatability, but it is difficult to guarantee the fidelity of experiments. On the contrary, hardware-based testbeds can guarantee fidelity, but in large-scale CPS testing, fully hardware-based testbeds are hard to implement and have a high cost. In addition, in the extreme case where a destructive or robust experiment is required,

TABLE 6. Communication infrastructure of cyber-physical system testbed.

Testbed	Communication Protocols	Technology	Network	Wireless/Wired
IPv6-Based Testbed[93]	IPv6	-	LAN	Wireless
Real-time Cyber-Physical Co-Simulation Testbed[94][95]	IEEE C37.118-2005, IEEE C37.118-2011	Ethernet	WAN	-
Testbed for UAV[96]	-	Ad hoc	-	Wireless
EPIC[97][98]	Modbus, DNP3, MMS protocols	Ethernet , Remote Procedure Call (RPC)	LAN, WAN	Wireless
UPBOT[99][100]	Serial Command Interface (SCI) protocol, TCP/IP	Socket	-	Wireless，Wired
Worcester Polytechnic Institute[101]	HTTP	MIMO, Bluetooth 4.0, WiFi	LAN	Wireless
RCPS testbed[102]	Openflow	Ethernet	-	Wired
FACIES Testbed[103][104]	Modbus, TCP / IP	-	-	-
CPSTCS Testbed[105]	Openflow, OF protocol	Cloud Computing, SDN	-	Wireless, Wired
SURE[106]	-	Cloud	-	-
PowerCyber Testbed[107][108]	IEC61850, C37.118, Modbus, DNP3.0, OPC UA	Ethernet	WAN, LAN	Wired
Turtlebot Robots Testbed[109]	IEEE 802.15.4	-	-	-
WSU Testbed[110]	OPC UA, IEC 61850, Multi Media Service protocol, DNPi, ICCP, UDP, TCP, DNP3.0	Ethernet	WAN	Wired
Open Heterogeneous Wireless Testbed[111]	IEEE 802.11, IEEE 802.15.4	WiFi, Ad-Hoc, Ethernet, VM	Internet	Wireless，Wired
Cloud Communication -Based Remote Access Testbed[112]	OPC UA, IEEE Std C37.118.1, IEEE Std C37.118.2, IEC 61850, Modbus	Ethernet	WAN	Wired
Microgrids Testbed[113]	Modbus	Ethernet	WAN, LAN, Internet	-
Multi-Agent Testbed[114]	Zigbee	WIFI, 4G, GPS	-	Wireless
Florida State University[115]	TCP / IP	Ethernet	WAN	Wired
GISOO[116]	IEEE 802.15.4	-	-	Wireless
Up and Away[117]	-	WiFi, Socket, GPS, Ad-Hoc	-	Wireless
INVITED[118]	IEEE 1588 Precision Time Protocol	-	LAN	Wireless
CPMC[119]	MTConnect, TCP / IP, REST	Cloud. Web	Internet	Wireless
Multifunctional CPS testbed[120]	DNP3.0, IEC60870-5-104	Ethernet	WAN	Wireless
South Dakota State University[121]	DNP3.0, SEL-C662	-	WAN	Wireless
Drone-Be-Gone[122]	-	-	LAN	Wireless
WADI[123]	RS485-Modbus/TCP	-	LAN	Wired

TABLE 7. Simulation tools of simulation-based testbed.

Reference	CPS Domains	Simulation Tools
[94][95]	Smart Grid	RTDS、RSCAD、DeterLab、NS-3
[102]	Distributed and Resilient CPS	Physical Simulators、Communication Network Emulator
[106]	Transportation Networks	Command-and-Control Wind Tunnel（C2WT）、WebGME
[107][108]	Smart Grid	RTDS、DigSilent
[109]	Autonomous Robots	Stage、OMNeT ++、sCPS（DEECo）
[111]	CPS	Software Tools
[113]	Smart Grid	Simulink、OPAL-RT、OMNeT++
[115]	Smart Grid	RTDS
[116]	Wireless CPS	Simulink、COOJA
[120]	Smart Grid	RTDS, WANE
[121]	Smart Grid	OPAL-RT, RT-lab

hardware-based testbeds can't guarantee complete safety. Therefore, the hybrid platform, which can benefit from the advantages of both, may attract more attention in future research.

The Furthermore, based on the execution mechanism, the architecture of CPS testbeds can also be divided into two categories according to their deployment: centralized or distributed. The centralized testbed concentrates all components of a system in a tightly coupled chassis and performs data acquisition and test analysis locally. For instance, the testbeds for smart grids [94], [95], distributed resilient CPSs [102], wireless CPSs [116], water systems [123], and UAVs [117] all integrate their components in a relatively small chassis. There is also a special testbed, where all of its components are implemented through programming in a computer simulation [106], [109]; we consider this as a centralized testbed. Distributed testbed refers to a platform that contains multiple connected devices, each of them conduct a different part of the same task, running simultaneously and working harmoniously under the control of people or control equipment. At present, most testbeds are distributed. Distributed testbeds are used in home equipment [93], critical infrastructure [97], [98], [105], robot systems [99]–[101], water systems [103], smart grids [107], [108], [112], [113], [115], [120], [121], general CPSs [111], [118], cloud manufacturing CPSs [119], and traffic systems [114]. Finally, in the literature [96], a testbed is considered centralized if the algorithm runs on the ground station of the testbed and is considered distributed if the algorithm runs on each UAV. Therefore, a testbed can be configured as either centralized or distributed for different testing requirements.

This condition also applies to [122], where the testbed can be treated as centralized or distributed. As mentioned above, most CPS testbeds are designed as distributed. The major advantage of centralized testbeds is their ease of use and even carry-on. However, due to the tightly coupled architecture, upgrades and functional expansion will be limited. Compared with centralized testbeds, the layout of distributed testbeds is more flexible to configuration for different testing requirements.

B. FUNCTION ANALYSIS OF CPS TESTBEDS

There are extensive studies on testbeds for CPSs. Each testbed has its own unique features and provides different kinds of functions. In this section, we classify these testbeds according to their most significant features and main functions: security-oriented, control-oriented, performance-oriented testbed, and Multi-objective comprehensive CPS testbed.

1) SECURITY-ORIENTED CPS TESTBEDS

Many CPS testbeds focus on forms of security, such as cyber security, communications security, and physical system security. Many security-oriented testbeds are developed for smart grids. Liu *et al.* [94] and Liu and Srivastava [95] developed a testbed that can be used to test the impact of cyber-attacks on smart grids, such as denial-of-service attacks and man-in-the-middle attacks, by modeling and simulating the power system through RTDS and RSCAD (RTDS simulator software). Network Simulator-3 (NS-3) and DeterLab can be used to simulate the communication network, and the ''Real-Time Voltage Stability Monitoring and Control (RT-VSMAC)'' tool is used as an application layer. Aditya Ashok *et al.* [107] and Hahn *et al.* [108] built the testbed called PowerCyber to test the impact of cyber-attacks on smart grids, which implements RTDS and Internet-scale event and attack generation environment (ISEAGE) WAN simulation. WSU Testbed [110] focuses on the cyber security of a smart grid's communication protocols. Its architecture includes ICT modules that cover multiple communication protocols, the power system simulation tool to model the behavior of physical power, the cyber module with SCADA, and the attack module for implementing a cyber-attack on the power system. Zhang *et al.* [120] developed a multifunctional testbed that contains real-time physical system simulation, network emulation, and multi-level control simulation for testing the cyber security of smart grid; this testbed was also integrated with a WAN emulator to provide advanced attack simulation. Poudel *et al.* [121] developed a real-time testbed for testing the cyber security and stability control of smart grids. OPAL-RT and RT-lab are used to simulate a real power system as well as a cyber system containing multiple substations and a control center; the two systems are communicate through DNP3.0.

There are also some security-oriented testbeds for other CPS domains. EPIC [97], [98] is a simulation based testbed for evaluating the impact of cyber-attacks on networked critical infrastructures (NCI). The physical process model is initially built into Matlab Simulink, and then automatically generated and integrated into the software simulation unit (SSim). Gao *et al.* [105] developed a testbed based on cloud computing and a software defined network (SDN) to test the impact of cyber threats on the cyber and physical dimensions of critical infrastructures. UPBOT [99], [100] is a testbed for testing the cyber security of robot systems, which includes the body, nerves, and brains. Miciolino *et al.* [103] built a water system testbed for testing the impact of cyber-attacks on the SCADA communications network; they particularly focused on the security of Modbus/TCP protocol. The Water distribution testbed (WADI) [123] is mainly used for detecting cyber-attacks and physical attacks and analyzing the cascading effects of attacks. Neema *et al.* [106] build a testbed called SURE by integrating a modeling and simulation platform called command-and-control wind tunnel (C2WT) and a web-based collaborative modeling tool named WebGME to test and evaluate the security and resilience of traffic networks.

2) CONTROL-ORIENTED CPS TESTBEDS

Control-oriented testbeds are very important for assuring the correctness of control logic for complex CPSs. The IPv6-based testbed [93] is built with laptops and smartphones for testing the remote dynamic control of smart home devices. Dimitrov *et al.* [101] developed a testbed for testing the control algorithm of home-assisted robots. Jamshidi *et al.* [96] developed a testbed for testing the co-control algorithm of swarm of UAV, which includes operator, ground station, and one block called *n* UAVs. The testbed Up and Away [117] is designed to abstract the control of the physical components in order to reduce the complexity of UAV-oriented CPS experiments. Drone-Be-Gone (DbeG) [122] is a UAV testbed that has the ability to switch between centralized or distributed control. DbeG has four main features: vision-based 2-D localization, autonomous navigation for multiple UAVs, simulation environment of testbed, and external processing units (EPUs). The Turtlebot Robots testbed [109] abstracts a robot as an autonomous component to test control algorithms for experimenting on, comparing, and developing new adaptive solutions related to intelligent CPSs. Cintuglu and Mohammed [112] developed a testbed to test the collaborative communication and control of an actual vehicle system. The architecture of this testbed includes a quadrotor vehicle, miniature vehicles, and high-performance embedded roadside units (RSUs).

3) PERFORMANCE-ORIENTED CPS TESTBEDS

Time is critical for CPSs, but the specification and verification of timing requirements for a CPS are costly and difficult to attain. The Microgrids testbed [113] is developed to test the impact of network delay on system performance. This testbed consists of three main components: real-time models of a distribution feeder model, NSIL models, and physical hardware. RCPS [102] is a testbed that simulates real

CPS deployment. It consists of 32 RCPS nodes connected to a programmable OpenFlow communication network switch through its Gigabit Ethernet port. This testbed can be used to test and analyze hardware and software fault tolerance, software reconfiguration, and system stability in design-time before deployment of a CPS. Szczodrak *et al.* [111] built a testbed for deploying, testing, reconfiguring, and evaluating applications and supporting firmware of CPSs; it consists of three main parts: a server back-end, backbone network, and testbed management unit. The INVITED [118] testbed is designed to test the timing behavior of CPSs. INVITED contains K nodes, and each node monitors the required signals and captures events using hardware timestamping; all nodes are synchronized through a reference clock over PTP to within microsecond accuracy.

4) MULTI-OBJECTIVE COMPREHENSIVE CPS TESTBEDS

Besides the categories described above, there are some other comprehensive testing platforms for CPSs with multi-objectives. Florida State University testbed is composed of real-time power system simulators [115]. It is based on real-time digital simulator (RTDS) technology with fiber and Ethernet networks to test the ''smart'' and distributed management control. Aminian *et al.* [116] built a testbed named GISOO, which integrates the Simulink and COOJA simulation tools; it can be used to comprehensively analyze computing, communication, and control components and their interaction on wireless CPSs. Liu *et al.* [119] developed a cyber-physical manufacturing cloud (CPMC), which is a new paradigm that integrates cloud manufacturing and CPSs for monitoring machining operations and performing manufacturing operations directly from a manufacturing cloud. They implemented a CPMC testbed containing two manufacturing sites connected to the cloud over the Internet. In summary, these testbeds always combine different kinds of testing capabilities together, but the development of multipurpose testbeds is still a task that should be further studied in the future.

V. CHALLENGES AND FUTURE RESEARCH FOR CPS TESTING

Studies on CPS testing have achieved great progress, but many issues remain, particularly due to the rapidly increasing complexity of CPSs. Based on the aforementioned analysis, this section will focus on the open challenges faced in CPS testing from the perspectives of testing methods and testbeds. We will also consider possible future works to provide useful insights for in-depth CPS research.

A. CHALLENGES FOR FUTURE COMPLEX CPS TESTING METHODS

Though many advanced testing methods for CPSs are rapidly developing, there remains several bottlenecks, especially for complex CPSs with high reliability, security, and resilience requirements. These challenges include state space explosion, uncertainty modeling, real-time assurance, and testing oracles.

- **Uncertainty modelling for CPS testing:** Uncertainty is ubiquitous in CPS due to the complex time-varying interactions between physical systems, network equipment, and computing infrastructure. Determining how to model these known and unknown uncertainties is one of the great challenges to overcome, especially for the online test generation, execution, and analysis [36].
- **State space explosion:** State space explosion is a classical long-term problem involved in both testing and formal verification, especially for the increasingly complex CPSs [130], [134], [135]. Testing systems with less inputs, high coverage, and efficient bug-triggering will almost certainly be the main objective of many different advanced testing methods in the future. Distributed model checking integrated with intelligent testing methods may be one interesting research direction [134].
- **Real-time CPS testing assurance:** Due to the communication delays, complex interdependencies between different components, and impact of fault propagation through a whole CPS, high real-time assurance for parallel testing of CPSs is important but difficult challenge for future CPS research 51], [73], [116], [118], [120].
- **Complex oracles for CPS testing.** Testing oracles is bottleneck for many kinds of testing, whether it is past, present, or future [44]. Traditional testing and formal verification techniques both struggle to analyze complex CPS hybrid behavior. Scalable and distributed runtime monitoring analysis, which can be dynamically evolved by advanced deep-learning techniques or GANs during the testing process, could be employed as a possible solution in the future.

Clearly, abovementioned challenges highlight opportunities for future research. Here, we give some possible solutions to inspire new research directions.

- **New test-driven paradigm for complex CPS testing** With the fast development of new deep-learning and data mining techniques, the process of complex CPS testing can be driven by new testing paradigms. Traditional search-based testing methods can be reinforced by integrating model-based testing with data-driven testing methods and fault injection techniques, which can increase the controllability and observability for complex CPS testing significantly [60].
- **New test execution mechanism for hybrid CPSs.** Matched with the increasing high configurability and variability of complex CPSs, cloud-based parallel testing mechanisms are will be import in the future of CPS testing [67]. It is also necessary to simulate and co-simulate many parallel physical processes to analyze the continuous interaction of the user, controller, and physical environment and to automatically identify the

worst-case scenarios. This will guide the generation of efficient test cases.

- **Big data analysis methods for in-depth CPS testing.** Many high-dimensional and multi-source heterogeneous data will be generated during the testing process for CPSs. Mining models from these big data through advanced deep-learning techniques, while considering the complexity of time constraints, spatial constraints, and dynamic continuous behavior constraints, is one of the interesting directions for future data-driven CPS testing. This can lead to tremendous potential applications [58]–[60], [62].
- **Combined schema for non-functional CPS testing**. The research on reliability testing and fragility testing for CPSs has just begun [4], 92]. Each of the single testing methods is difficult to achieve for complex CPSs, but the non-functional CPS properties are in fact closely related to each other in essence, sharing some enabling testing techniques (e.g., fault injection). Therefore, exploring the possibility of combining different functional and non-functional CPS testing methods efficiently during one system-level testing process is an interesting prospect for future research. Comprehensively evaluating the reliability, fragility, and elasticity of CPSs in this new testing paradigm is another possible new research direction.

B. KEY FUTURE RESEARCH ISSUES FOR CPS TESTBEDS

There is no comprehensive and universal testbed for all CPSs, and developing a scalable testbed for complex CPSs still holds many challenges. Based on the ideas formulated in the previous section, it is desirable for future CPS testbeds to improve the following characteristics:

- Accuracy: CPS testbeds should be able to monitor the testing process accurately with high time and value precision, especially in complex environments filled with uncertain external or internal noise interferences [11], [36].
- Automation: CPS testing is time consuming and labor intensive. One of the most important issues for testbed improvement is automatic testing. Considering the complex constraint faced by complex interaction between software, hardware, and networks, automating the generation and execution of suitable test cases is very important and is a great challenge [14].
- Controllability and Observability: Though many current CPS testbeds are decentralized, in the future, more attention should be focused on networked testbeds to ensure accurate, fast, and complete information processing with friendly interaction. High controllability and observability are more desirable for the whole CPS testing process [48], [102].
- Reliability and Reproducibility: CPS testing must be repeatable to obtain consistent results. Therefore, reliable CPS testbeds must be able to schedule and control all events during the testing process with certain reproducibility [4].
- Safe execution: In some extreme case where a destructive fault injection is required for a physical component, it may endanger the safety of the testbed or the CPS under testing [14], [22], [49]. Hence, ensuring the safe execution of responsible automatic CPS testing without self-destructiveness in any situation is a challenge for future research.
- High speed and capacity: The generation and execution of test cases for higher coverage of complex CPSs is becoming inevitably time consuming. The corresponding testing data generated is also growing in volume. It is evident that future testbeds for CPSs must be further optimized with fast big data analysis capacity [14], [20].

In addition, to overcoming these challenges faced by existing testbeds, we believe that future research should include the following objectives:

- Ultra-large scale: Due to the advancement of complex system theories, many new properties are emerging when the system reaches a certain scale and complexity level. Most current testbeds are only feasible for small or medium size CPSs [14]. Therefore, future works should aim to expand the testbeds suitable for ultra-complex CPS testing.
- Multiple attack scenarios: Fundamentally, most existing testbeds that focus on security testing can only implement some common cyber-attack scenarios for a specific CPS (such as smart grid) [108]. In the future, it is necessary to develop a testbed that can implement multiple attack scenarios for multiple different systems at the same time [14].
- Multi-objective: As far as we know, there is no testbed that can test the reliability, resilience, fragility, and robustness of a CPS comprehensively [106]. Apparently, in the future, developing a testbed for efficient multi-objective testing, including simultaneous functional and non-functional testing, will be an active research area. This can be achieved by integrating advanced emerging technologies, such as IoT, big data, cloud computing, and AI [119], [120].

VI. CONCLUSIONS

CPS testing is still a challenging research field due to the increasing heterogeneity, scale, and complexity. In this paper, we give a comprehensive overview of state-of-the-art CPS testing methods and testbeds. Although CPSs bring some advances for the existing testing theory and technology, there are still many limitations for the wider industrial application of CPS testing. Accordingly, by considering some additional requirements and constraints, we highlight challenges faced by existing test methods and testbeds, and by doing so, we formulate possible future research directions. The analysis and discussion in this paper can provide useful insights for

CPS researchers to instantiate different CPS testing inventions or new applications.

REFERENCES

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

[2] E. A. Lee, "Cyber physical systems: Design challenges," in *Proc. IEEE Symp. Object Oriented Real-Time Distrib. Comput.*, May 2008, pp. 363–369.

[3] G. A. Fink, T. W. Edgar, T. R. Rice, D. G. Macdonald, and C. E. Crawford, "Security and privacy in cyber-physical systems," in *Cyber-Physical Systems*. Amsterdam, The Netherlands: Elsevier, 2017, ch. 9, pp. 129–141.

[4] Z. Li and R. Kang, "Strategy for reliability testing and evaluation of cyber physical systems," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage.*, Dec. 2016, pp. 1001–1006.

[5] H. Zihe and K. Rui, *Testing and Evaluation Technology of Cyber-Physical System*, 1st ed. Beijing, China: Posts & Telecom Press, 2016, p. 22.

[6] S. Tobuschat, R. Ernst, A. Hamann, and D. Ziegenbein, "System-level timing feasibility test for cyber-physical automotive systems," in *Proc. Ind. Embedded Syst.*, 2016, pp. 1–10.

[7] T. M. Chen, J. C. Sanchez-Aarnoutse, and J. Buford, "Petri net modeling of cyber-physical attacks on smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 741–749, Dec. 2011.

[8] R. Sinha, C. Pang, G. S. Martinez, J. Kuronen, and V. Vyatkin, "Requirements-aided automatic test case generation for industrial cyber-physical systems," in *Proc. Int. Conf. Eng. Complex Comput. Syst.*, 2016, pp. 198–201.

[9] E. Sultanovs and A. Romānovs, "Centralized healthcare cyber-physical system's data analysis module development," in *Proc. IEEE 4th Workshop Adv. Inf., Electron. Elect. Eng. (AIEEE)*, Nov. 2016, pp. 1–4.

[10] C. Farrar and D. Mascareñas, "A preliminary cyber-physical security assessment of the robot operating system (ROS)," *Proc. SPIE*, vol. 8741, no. 3, pp. 292–298, May 2013.

[11] V. Gunes, S. Peter, T. Givargis, and F. Vahid, "A survey on concepts, applications, and challenges in cyber-physical systems," *KSII Trans. Internet Inf. Syst.*, vol. 8, no. 12, pp. 4242–4268, 2014.

[12] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, "A survey on platoon-based vehicular cyber-physical systems," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 263–284, 1st Quart., 2016.

[13] M. H. Cintuglu, O. A. Mohammed, K. Akkaya, and A. S. Uluagac, "A survey on smart grid cyber-physical system testbeds," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 446–464, 1st Quart., 2017.

[14] S. A. Asadollah, R. Inam, and H. Hansson, "A survey on testing for cyber physical system," in *Proc. IFIP Int. Conf. Test. Softw. Syst.*, 2015, pp. 194–207.

[15] W. U. Guangyu, J. Sun, and J. Chen, "A survey on the security of cyber-physical systems," *Control Theory Technol.*, vol. 14, no. 1, pp. 2–10, 2016.

[16] R. F. Babiceanu and R. Seker, "Big Data and virtualization for manufacturing cyber-physical systems: A survey of the current status and future outlook," *Comput. Ind.*, vol. 81, pp. 128–137, Sep. 2016.

[17] Q. Shafi, "Cyber physical systems security: A brief survey," in *Proc. Int. Conf. Comput. Sci. Appl.*, 2012, pp. 146–150.

[18] R. Chaâri *et al.*, "Cyber-physical systems clouds: A survey," *Comput. Netw.*, vol. 108, pp. 260–278, Oct. 2016.

[19] 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.

[20] S. K. Khaitan and J. D. McCalley, "Design techniques and applications of cyberphysical systems: A survey," *IEEE Syst. J.*, vol. 9, no. 2, pp. 350–365, Jun. 2015.

[21] S. A. Haque, S. M. Aziz, and M. Rahman, "Review of cyber-physical system in healthcare," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 4, p. 217415, 2014.

[22] Y. Liu, Y. Peng, B. Wang, S. Yao, and Z. Liu, "Review on cyber-physical systems," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 1, pp. 27–40, Jan. 2017.

[23] M. Wenfang, "CPS: Sensor-net to sensor-acuator-net," *China Inf. World*, vol. 25, no. 41, pp. 310–315, 2010.

[24] H. Zihe and K. Rui, *Testing and Evaluation Tecgnology of Cyber-Physical System*. Beijing, China: Posts & Telecom Press, 2016, p. 8.

[25] Y. Tan, S. Goddard, and L. C. Pérez, "A prototype architecture for cyber-physical systems," *ACM SIGBED Rev.*, vol. 5, no. 1, pp. 1–2, 2008.

[26] R. Rajkumar, "A cyber–physical future," *Proc. IEEE*, vol. 100, Special Centennial Issue, pp. 1309–1312, May 2012.

[27] I. Schieferdecker, "Model-based testing," *IEEE Softw.*, vol. 29, no. 1, pp. 14–18, Jan./Feb. 2012.

[28] I. K. El-Far and J. A. Whittaker, *Model-Based Software Testing*. Hoboken, NJ, USA: Wiley, 2002.

[29] L. C. Silva, M. Perkusich, F. M. Bublitz, H. O. Almeida, and A. Perkusich, "A model-based architecture for testing medical cyber-physical systems," in *Proc. 29th Annu. ACM Symp. Appl. Comput.*, 2014, pp. 25–30.

[30] Z. Jiang, M. Pajic, and R. Mangharam, "Cyber–physical modeling of implantable cardiac medical devices," *Proc. IEEE*, vol. 100, no. 1, pp. 122–137, Jan. 2011.

[31] J. Zander, "Model-based testing for execution algorithms in the simulation of cyber-physical systems," in *Proc. IEEE AUTOTESTCON*, Sep. 2013, pp. 1–7.

[32] F. Saglietti, D. Föhrweiser, S. Winzinger, and R. Lill, "Model-based design and testing of decisional autonomy and cooperation in cyber-physical systems," in *Proc. 41st Euromicro Conf. Softw. Eng. Adv. Appl.*, 2015, pp. 479–483.

[33] I. Buzhinsky, C. Pang, and V. Vyatkin, "Formal modeling of testing software for cyber-physical automation systems," in *Proc. Trustcom/BigDataSE/ISPA*, 2015, pp. 301–306.

[34] A. M. Kosek and O. Gehrke, "Ensemble regression model-based anomaly detection for cyber-physical intrusion detection in smart grids," in *Proc. EPEC*, Oct. 2016, pp. 1–7.

[35] A. Aerts, M. R. Mousavi, and M. Reniers, "A tool prototype for model-based testing cyber-physical systems," in *Proc. Int. Colloq. Theor. Aspects Comput.*, 2015, pp. 563–572.

[36] S. Ali and T. Yue, "U-test: Evolving, modelling and testing realistic uncertain behaviours of cyber-physical systems," in *Proc. IEEE Int. Conf. Softw. Test., Verification Validation*, Apr. 2015, pp. 1–2.

[37] A. Schmidt, U. Durak, and T. Pawletta, "Model-based testing methodology using system entity structures for MATLAB/simulink models," *Simulation*, vol. 92, no. 8, pp. 729–746, 2016.

[38] A. Motii, A. Lanusse, B. Hamid, and J. M. Bruel, "Model-based real-time evaluation of security patterns: A SCADA system case study," in *Proc. Int. Conf. Comput. Saf., Rel., Secur.*, 2016, pp. 375–389.

[39] H. Orojloo and M. A. Azgomi, "A game-theoretic approach to model and quantify the security of cyber-physical systems," *Comput. Ind.*, vol. 88, pp. 44–57, Jun. 2017.

[40] A. Arrieta, S. Wang, G. Sagardui, and L. Etxeberria, "Search-based test case selection of cyber-physical system product lines for simulation-based validation," in *Proc. Int. Syst. Softw. Product Line Conf.*, 2016, pp. 297–306.

[41] A. Arrieta, S. Wang, G. Sagardui, and L. Etxeberria, "Test case prioritization of configurable cyber-physical systems with weight-based search algorithms," in *Proc. Genetic Evol. Comput. Conf.*, 2016, pp. 1053–1060.

[42] R. Matinnejad, S. Nejati, L. Briand, T. Bruckmann, and C. Poull, "Search-based automated testing of continuous controllers: Framework, tool support, and case studies," *Inf. Softw. Technol.*, vol. 57, no. 1, pp. 705–722, 2015.

[43] T. Yue, S. Ali, and K. Nie, "Towards a search-based interactive configuration of cyber-physical system product lines," in *Proc. CEUR*, 2013, pp. 71–75.

[44] A. Kane, T. Fuhrman, and P. Koopman, "Monitor based oracles for cyber-physical system testing: Practical experience report," in *Proc. 44th Annu. IEEE/IFIP Int. Conf. Dependeble Syst. Netw.*, Jun. 2014, pp. 148–155.

[45] J. Mao and L. Chen, "Runtime monitoring for cyber-physical systems: A case study of cooperative adaptive cruise control," in *Proc. 2nd Int. Conf. Intell. Syst. Design Eng. Appl.*, 2012, pp. 509–515.

[46] X. Zheng, C. Julien, R. Podorozhny, F. Cassez, and T. Rakotoarivelo, "Efficient and scalable runtime monitoring for cyber–physical system," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1667–1678, Jun. 2016.

[47] C. Chen, J. Yan, N. Lu, Y. Wang, X. Yang, and X. Guan, "Ubiquitous monitoring for industrial cyber-physical systems over relay-assisted wireless sensor networks," *IEEE Trans. Emerg. Topics Comput.*, vol. 3, no. 3, pp. 352–362, Sep. 2015.

[48] M. Mehrabian *et al.*, "Timestamp temporal logic (TTL) for testing the timing of cyber-physical systems," *ACM Trans. Embedded Comput. Syst.*, vol. 16, no. 5, pp. 1–20, 2017.

[49] A. Faza, S. Sedigh, and B. Mcmillin, "Integrated cyber-physical fault injection for reliability analysis of the smart grid," in *Proc. Int. Conf. Comput. Saf., Rel., Secur.*, 2010, pp. 277–290.

[50] J. Frohlich, J. Frtunikj, S. Rothbauer, and C. Stuckjurgen, "Testing safety properties of cyber-physical systems with non-intrusive fault injection—An industrial case study," in *Computer Safety, Reliability, and Security* (Lecture Notes in Computer Science), vol. 9923, A. Skavhaug, J. Guiochet, E. Schoitsch, and F. Bitsch, Eds. Cham, Switzerland: Springer, 2016, pp. 105–117.

[51] B. Vedder, T. Arts, J. Vinter, and M. Jonsson, "Combining fault-injection with property-based testing," in *Proc. Int. Workshop Eng. Simulations Cyber-Phys. Syst.*, 2013, pp. 1–8.

[52] H. Orojloo and M. A. Azgomi, "A method for evaluating the consequence propagation of security attacks in cyber–physical systems," *Future Gener. Comput. Syst.*, vol. 67, pp. 57–71, Feb. 2016.

[53] J. Lee *et al.*, "Industrial big data analytics and cyber-physical systems for future maintenance & service innovation," in *Proc. 4th Int. Conf. Through-Life Eng. Services*, vol. 38, R. Roy, Eds. Amsterdam, The Netherlands: Elsevier, 2015, pp. 3–7.

[54] L. Zhang, "Designing big data driven cyber physical systems based on AADL," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2014, pp. 3072–3077.

[55] A. Marini and D. Bianchini, "Big data as a service for monitoring cyber-physical production systems," in *Proc. ECMS*, 2016, pp. 579–586.

[56] S. Don and M. Dugki, "Medical cyber physical systems andbigdata platforms," in *Proc. Med. Cyber Phys. Syst. Workshop*, Philadelphia, PA, USA, 2013, pp. 1–5.

[57] L. Wang and G. Wang, "Big data in cyber-physical systems, digital manufacturing and industry 4.0," *Int. J. Eng. Manuf.*, vol. 6, no. 4, pp. 1–8, 2016.

[58] C. K. M. Lee, C. L. Yeung, and M. N. Cheng, "Research on IoT based cyber physical system for industrial big data analytics," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage.*, Dec. 2015, pp. 1855–1859.

[59] J. Lee, B. Bagheri, and H. A. Kao, "Recent advances and trends of cyber-physical systems and big data analytics in industrial informatics," in *Proc. Int. Conf. Ind. Inform.*, 2014, pp. 1–6.

[60] O. Niggemann, G. Biswas, J. S. Kinnebrew, H. Khorasgani, S. Volgmann, and A. Bunte, "Data-driven monitoring of cyber-physical systems leveraging on big data and the Internet-of-Things for diagnosis and control," in *Proc. DX Safeprocess*, 2015, pp. 185–192.

[61] A. J. Jara, D. Genoud, and Y. Bocchi, "Big data for cyber physical systems: An analysis of challenges, solutions and opportunities," in *Proc. 8th Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput.*, 2014, pp. 376–380.

[62] W. Zhong and L. Zhang, "Challenges of big data based cyber-physical system," presented at the 2nd Workshop Adv. Res. Technol. Ind. Appl., Dalian, China, May 2016, Paper G0772.

[63] Y. W. Ahn and A. M. K. Cheng, "Automatic resource scaling for medical cyber-physical systems running in private cloud computing architecture," in *Proc. MCPS CPSWeek*, vol. 36. Wadern, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. Dagstuhl, 2014, pp. 58–65.

[64] Y. Zhang, M. Qiu, C.-W. Tsai, M. M. Hassan, and A. Alamri, "Health-CPS: Healthcare cyber-physical system assisted by cloud and big data," *IEEE Syst. J.*, vol. 11, no. 1, pp. 88–95, Mar. 2015.

[65] H. Abid, L. T. T. Phuong, J. Wang, S. Lee, and S. Qaisar, "V-cloud: vehicular cyber-physical systems and cloud computing," in *Proc. 4th Int. Symp. Appl. Sci. Biomed. Commun. Technol.*, 2011, p. 165.

[66] V. Hahanov *et al.*, "Cyber physical system—Smart cloud traffic control," in *Proc. Design Test Symp. (EWDTS)*, 2014, pp. 1–18.

[67] C. Berger, "Cloud-based testing for context-aware cyber-physical systems," in *Software Testing in the Cloud: Perspectives on an Emerging Discipline*. Hershey, PA, USA: IGI Global, 2012, pp. 68–95.

[68] J. Puttonen, S. O. Afolaranmi, L. G. Moctezuma, A. Lobov, and J. L. M. Lastra, "Enhancing security in cloud-based cyber-physical systems," *J. Cloud Comput. Res.*, vol. 2, no. 1, pp. 18–33, 2016.

[69] Y. B. Reddy, "Cloud-based cyber physical systems: Design challenges and security needs," in *Proc. 10th Int. Conf. Mobile Ad-Hoc Sensor Netw. (MSN)*, 2014, pp. 315–322.

[70] M. Azab and M. Eltoweissy, "Defense as a service cloud for cyber-physical systems," in *Proc. 7th Int. Conf. Collaborative Comput., Netw., Appl. Worksharing (CollaborateCom)*, 2011, pp. 392–401.

[71] S. Karnouskos, A. W. Colombo, and T. Bangemann, "Trends and challenges for cloud-based industrial cyber-physical systems," in *Industrial Cloud-Based Cyber-Physical Systems*. Cham, Switzerland: Springer, 2014, pp. 231–240.

[72] S. Karnouskos *et al.*, "The IMC-AESOP architecture for cloud-based industrial cyber-physical systems," in *Industrial Cloud-Based Cyber-Physical Systems*. Cham, Switzerland: Springer, 2014, pp. 49–88.

[73] K. Nakauchi, F. Bronzino, Y. Shoji, I. Seskar, and D. Raychaudhuri, "vMCN: Virtual mobile cloud network for realizing scalable, real-time cyber physical systems," in *Proc. Workshop Distrib. Cloud Comput. (DCC)*, 2016, p. 2.

[74] K. M. Alam and A. El Saddik, "C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems," *IEEE Access*, vol. 5, pp. 2050–2062, 2017.

[75] Z. Shu, J. Wan, D. Zhang, and D. Li, "Cloud-integrated cyber-physical systems for complex industrial applications," *Mobile Netw. Appl.*, vol. 21, no. 5, pp. 865–878, Oct. 2016.

[76] H. Y. Abbas, "Test-based falsification and conformance testing for cyber-physical systems," Ph.D. dissertation, Dept. Elect. Eng., Arizona State Univ., Tempe, AZ, USA, 2015.

[77] H. Abbas, B. Hoxha, G. Fainekos, J. V. Deshmukh, J. Kapinski, and K. Ueda. (2014). "Conformance testing as falsification for cyber-physical systems." [Online]. Available: https://arxiv.org/abs/1401.5200

[78] M. Mohaqeqi and M. R. Mousavi, "Sound test-suites for cyber-physical systems," in *Proc. 10th Int. Symp. Theor. Aspects Softw. Eng. (TASE)*, 2016, pp. 42–48.

[79] S. Li, X. Chen, Y. Wang, and M. Sun, "A framework for off-line conformance testing of timed connectors," in *Proc. Int. Symp. Theor. Aspects Softw. Eng. (TASE)*, 2015, pp. 15–22.

[80] M. Woehrle, K. Lampka, and L. Thiele, "Segmented state space traversal for conformance testing of cyber-physical systems," in *Proc. Int. Conf. Formal Modeling Anal. Timed Syst.* Berlin, Germany: Springer, 2011, pp. 193–208.

[81] M. Woehrle, K. Lampka, and L. Thiele, "Conformance testing for cyber-physical systems," *ACM Trans. Embedded Comput. Syst.*, vol. 11, no. 4, p. 84, 2012.

[82] H. Abbas, G. Fainekos, S. Sankaranarayanan, F. Ivančić, and A. Gupta, "Probabilistic temporal logic falsification of cyber-physical systems," *ACM Trans. Embedded Comput. Syst.*, vol. 12, no. 2, p. 95, 2013.

[83] H. Abbas, B. Hoxha, G. Fainekos, and K. Ueda, "Robustness-guided temporal logic testing and verification for stochastic cyber-physical systems," in *Proc. IEEE 4th Annu. Int. Conf. Cyber Technol. Automat., Control, Intell. Syst. (CYBER)*, Jun. 2014, pp. 1–6.

[84] G. Fainekos *et al.*, "Robust testing for discrete-time and continuous-time system models," U.S. Patent Application 12/708,651, Nov. 25, 2010.

[85] S. Pan, T. H. Morris, and U. Adhikari, "A specification-based intrusion detection framework for cyber-physical environment in electric power system," *IJ Netw. Secur.*, vol. 17, no. 2, pp. 174–188, 2015.

[86] T. Liu *et al.*, "Abnormal traffic-indexed state estimation: A cyber–physical fusion approach for smart grid attack detection," *Future Gener. Comput. Syst.*, vol. 49, pp. 94–103, Aug. 2015.

[87] Y. Sun, X. Guan, T. Liu, and Y. Liu, "A cyber-physical monitoring system for attack detection in smart grid," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2013, pp. 33–34.

[88] X. Luo, J. Li, Z. Jiang, and X. Guan, "Complete observation against attack vulnerability for cyber-physical systems with application to power grids," in *Proc. 5th Int. Conf. Electr. Utility Deregulation Restruct. Power Technol. (DRPT)*, 2015, pp. 962–967.

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

[90] Y. Chen, S. Kar, and J. M. F. Moura, "Dynamic attack detection in cyber-physical systems with side initial state information," *IEEE Trans. Autom. Control*, vol. 62, no. 9, pp. 4618–4624, Sep. 2016.

[91] H. L. Gawand, A. Bhattacharjee, and K. Roy, "Online monitoring of a cyber physical system against control aware cyber attacks," *Procedia Comput. Sci.*, vol. 70, pp. 238–244, Jan. 2015.

[92] T. Ma, S. Ali, T. Yue, and M. Elaasar, "Fragility-oriented testing with model execution and reinforcement learning," in *Testing Software and Systems* (Lecture Notes in Computer Science), vol. 10533, N. Yevtushenko, A. Cavalli, and H. Yenigün, Eds. Cham, Switzerland: Springer, 2017.

[93] O. P. Sang, T. H. Do, Y.-S. Jeong, and S. J. Kim, "A dynamic control middleware for cyber physical systems on an IPv6-based global network," *Int. J. Commun. Syst.*, vol. 26, no. 6, pp. 690–704, 2013.

[94] R. Liu, C. Vellaithurai, S. S. Biswas, T. T. Gamage, and A. K. Srivastava, "Analyzing the cyber-physical impact of cyber events on the power grid," *IEEE Trans. Smart Grid*, vol. 6, no. 5, pp. 2444–2453, Sep. 2015.

[95] R. Liu and A. Srivastava, "Integrated simulation to analyze the impact of cyber-attacks on the power grid," in *Proc. Modeling Simulation Cyber-Phys. Energy Syst.*, 2015, pp. 1–6.

[96] M. Jamshidi, A. S. J. Betancourt, and J. Gomez, "Cyber-physical control of unmanned aerial vehicles," *Scientia Iranica*, vol. 18, no. 3, pp. 663–668, 2011.

[97] C. Siaterlis, B. Genge, and M. Hohenadel, "EPIC: A testbed for scientifically rigorous cyber-physical security experimentation," *IEEE Trans. Emerg. Topics Comput.*, vol. 1, no. 2, pp. 319–330, Dec. 2013.

[98] Y. Soupionis and T. Benoist, "Cyber-physical testbed—The impact of cyber attacks and the human factor," in *Proc. Int. Conf. Internet Technol. Secured Trans.*, 2015, pp. 326–331.

[99] T. L. Crenshaw and S. Beyer, "UPBOT: A testbed for cyber-physical systems," in *Proc. Int. Conf. Cyber Secur. Exp. Test*, 2010, pp. 1–8.

[100] T. L. A. Crenshaw, *Using Robots and Contract Learning to Teach Cyber-Physical Systems to Undergraduates*. Piscataway, NJ, USA: IEEE Press, 2013, pp. 116–120.

[101] V. Dimitrov, V. Jagtap, M. Wills, J. Skorinko, and T. Padır, "A cyber physical system testbed for assistive robotics technologies in the home," in *Proc. Int. Conf. Adv. Robot.*, 2015, pp. 323–328.

[102] P. S. Kumar, W. Emfinger, and G. Karsai, "A testbed to simulate and analyze resilient cyber-physical systems," in *Proc. Int. Symp. Rapid Syst. Prototyping*, 2015, pp. 97–103.

[103] E. E. Miciolino, G. Bernieri, F. Pascucci, and R. Setola, "Communications network analysis in a SCADA system testbed under cyber-attacks," in *Telecommun. Forum Telfor*, 2016, pp. 341–344.

[104] G. Bernieri, E. E. Miciolino, F. Pascucci, and R. Setola, "Monitoring system reaction in cyber-physical testbed under cyber-attacks," *Comput. Elect. Eng.*, vol. 59, pp. 86–98, Apr. 2017.

[105] H. Gao, Y. Peng, K. Jia, and Z. Wen, "Cyber-physical systems testbed based on cloud computing and software defined network," in *Proc. Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, 2015, pp. 337–340.

[106] H. Neema *et al.*, "Demo abstract: SURE: An experimentation and evaluation testbed for CPS security and resilience," in *Proc. Int. Conf. Cyber-Phys. Syst.*, 2016, p. 27.

[107] A. Ashok, P. Wang, M. Brown, and M. Govindarasu, "Experimental evaluation of cyber attacks on automatic generation control using a CPS security testbed," in *Proc. Power Energy Soc. Gen. Meeting*, 2015, pp. 1–5.

[108] A. Hahn, A. Ashok, S. Sridhar, and M. Govindarasu, "Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 847–855, Jun. 2013.

[109] V. Matena, T. Bures, I. Gerostathopoulos, and P. Hnetynka, "Model problem and testbed for experiments with adaptation in smart cyber-physical systems," in *Proc. IEEE/ACM Int. Symp. Softw. Eng. Adapt. Self-Manag. Syst.*, May 2017, pp. 82–88.

[110] C.-C. Sun, J. Hong, and C.-C. Liu, "A co-simulation environment for integrated cyber and power systems," in *Proc. IEEE Int. Conf. Smart Grid Commun.*, Nov. 2015, pp. 133–138.

[111] M. Szczodrak, Y. Yang, D. Cavalcanti, and L. P. Carloni, "An open framework to deploy heterogeneous wireless testbeds for cyber-physical systems," in *Proc. IEEE Int. Symp. Ind. Embedded Syst.*, Nov. 2013, pp. 215–224.

[112] M. H. Cintuglu and O. A. Mohammed, "Cloud communication for remote access smart grid testbeds," in *Proc. Power Energy Soc. Gen. Meeting*, Jul. 2016, pp. 1–5.

[113] A. Nelson *et al.*, "Cyber-physical test platform for microgrids: Combining hardware, hardware-in-the-loop, and network-simulator-in-the-loop," in *Proc. Power Energy Soc. Gen. Meeting*, Jul. 2016, pp. 1–5.

[114] Y. Zhou *et al.*, "Demo: The multi-agent based evaluation of connected vehicle systems," in *Proc. Veh. Netw. Conf.*, 2015, pp. 131–132.

[115] M. J. Stanovich, "Development of a smart-grid cyber-physical systems testbed," in *Proc. ISGT*, Feb. 2013, pp. 1–6.

[116] B. Aminian, J. Araújo, M. Johansson, and K. H. Johansson, "GISOO: A virtual testbed for wireless cyber-physical systems," in *Proc. IECON*, vol. 20, 2013, pp. 5588–5593.

[117] A. Saeed, A. Neishaboori, A. Mohamed, and K. A. Harras, "Up and away: A visually-controlled easy-to-deploy wireless UAV cyber-physical testbed," in *Proc. IEEE Int. Conf. Wireless Mobile Comput., Netw. Commun.*, Oct. 2014, pp. 578–584.

[118] A. Shrivastava *et al.*, "INVITED: A testbed to verify the timing behavior of cyber-physical systems," in *Proc. Design Automat. Conf.*, Jun. 2017, p. 69.

[119] X. F. Liu, M. R. Shahriar, S. M. N. Al Sunny, M. C. Leu, and L. Hu, "Cyber-physical manufacturing cloud: Architecture, virtualization, communication, and testbed," *J. Manuf. Syst.*, vol. 43, pp. 352–364, Apr. 2017.

[120] H. Zhang, D. Ge, J. Liu, and Y. Zhang, "Multifunctional cyber-physical system testbed based on a source-grid combined scheduling control simulation system," *IET Gener. Transmiss. Distrib.*, vol. 11, no. 12, pp. 3144–3151, 2017.

[121] S. Poudel, Z. Ni, and N. Malla, "Real-time cyber physical system testbed for power system security and control," *Int. J. Elect. Power Energy Syst.*, vol. 90, pp. 124–133, Sep. 2017.

[122] K. A. Harras, K. A. Harras, K. A. Harras, and K. A. Harras, "Simulating drone-be-gone: Agile low-cost cyber-physical UAV testbed (demonstration)," in *Proc. Int. Conf. Auton. Agents Multiagent Syst.*, 2016, pp. 1491–1492.

[123] C. M. Ahmed, V. R. Palleti, and A. P. Mathur, "WADI: A water distribution testbed for research in the design of secure cyber physical systems," in *Proc. Int. Workshop Cyber-Phys. Syst. Smart Water Netw.*, 2017, pp. 25–28.

[124] J. Kapinski, J. V. Deshmukh, X. Jin, H. Ito, and K. Butts, "Simulation-based approaches for verification of embedded control systems: An overview of traditional and advanced modeling, testing, and verification techniques," *IEEE Control Syst.*, vol. 36, no. 6, pp. 45–64, Dec. 2016.

[125] E. Bartocci, J. Deshmukh, A. Donzé, G. Fainekos, O. Maler, and D. Nickovic, "Specification-based monitoring of cyber-physical systems: A survey on theory, tools and applications," in *Lectures on Runtime Verification*. Cham, Switzerland: Springer, 2018, pp. 135–175.

[126] Y. Guo, X. Hu, B. Hu, J. Cheng, M. Zhou, and R. Y. K. Kwok, "Mobile cyber physical systems: Current challenges and future networking applications," *IEEE Access*, vol. 6, pp. 12360–12368, 2017.

[127] S. Mitsch and A. Platzer, "ModelPlex: Verified runtime validation of verified cyber-physical system models," in *Proc. Int. Conf. Runtime Verification*. New York, NY, USA: Springer, 2014, pp. 199–214.

[128] Q. Luo *et al.*, "RV-monitor: Efficient parametric runtime verification with simultaneous properties," in *Runtime Verification* (Lecture Notes in Computer Science), vol. 8734, B. Bonakdarpour and S. A. Smolka, Eds. Cham, Switzerland: Springer, 2014.

[129] X. Zheng, C. Julien, R. Podorozhny, and F. Cassez, "BraceAssertion: Runtime verification of cyber-physical systems," in *Proc. IEEE Int. Conf. Mobile Ad Hoc Sensor Syst.*, Oct. 2015, pp. 298–306.

[130] J. Mrázek, P. Bauch, H. Lauko, and J. Barnat, "SymDIVINE: Tool for control-explicit data-symbolic state space exploration," in *Model Checking Software*. Cham, Switzerland: Springer, 2016, pp. 208–213.

[131] K. Yu, Z. Chen, and W. Dong, "A predictive runtime verification framework for cyber-physical systems," in *Proc. IEEE 8th Int. Conf. Softw. Secur. Rel. Companion*, Jun./Jul. 2014, pp. 247–250.

[132] J. V. Deshmukh, A. Donzé, S. Ghosh, X. Jin, G. Juniwal, and S. A. Seshia, "Robust online monitoring of signal temporal logic," in *Runtime Verification*. New York, NY, USA: Springer, 2015, pp. 133–136.

[133] J. Huang *et al.*, "ROSRV: Runtime verification for robots," in *Proc. Int. Conf. Runtime Verification*, 2014, pp. 247–254.

[134] A. Desai, T. Dreossi, and S. A. Seshia, "Combining model checking and runtime verification for safe robotics," in *Runtime Verification* (Lecture Notes in Computer Science), vol. 10548, S. Lahiri and G. Reger, Eds. Cham, Switzerland: Springer, 2017.

[135] B. Andersson, S. Chaki, and D. de Niz, "Combining symbolic runtime enforcers for cyber-physical systems," in *Runtime Verification* (Lecture Notes in Computer Science), vol. 10548, S. Lahiri and G. Reger, Eds. Cham, Switzerland: Springer, 2017.

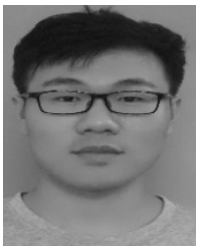
[136] Y. Annpureddy, C. Liu, G. Fainekos, and S. Sankaranarayanan, "S-TaLiRo: A tool for temporal logic falsification for hybrid systems," in *Tools and Algorithms for the Construction and Analysis of Systems*. Berlin, Germany: Springer, 2011.

[137] G. Fainekos, "Automotive control design bug-finding with the S-TaLiRo tool," in *Proc. Amer. Control Conf.*, 2015, p. 4096.

[138] D. Jin, P. O. Meredith, C. Lee, and G. Roşu, "JavaMOP: Efficient parametric runtime monitoring framework," in *Proc. Int. Conf. Softw. Eng.*, 2012, pp. 1427–1430.

XIN ZHOU (M'94) received the B.S. degree in reliability and systems engineering from Beihang University, Beijing, China, in 2016, where he is currently pursuing the Ph.D. degree with the School of Reliability and Systems Engineering.

His research interests include testing and reliability analysis for cyber physical system and complex network.



XIAODONG GOU (M'92) received the B.S. degree from the School of Management Engineering, Zhengzhou University of Aeronautics, Zhengzhou, China, in 2016. He is currently pursuing the Ph.D. degree with the School of Reliability and Systems Engineering, Beihang University, Beijing, China. His research interests include testing and diagnosis for complex system.

TINGTING HUANG received the Ph.D. degree in reliability and systems engineering from Beihang University, Beijing, China, in 2000. She is currently an Assistant Professor with the School of Reliability and Systems Engineering, Beihang University.

Her research interests include testing and quality analysis for software, hardware, and cyber physical system.

SHUNKUN YANG (M'78) received the B.S., M.S., and Ph.D. degrees in reliability and systems engineering from Beihang University, Beijing, China, in 2000, 2003, and 2011, respectively. He is currently an Associate Research Professor with the School of Reliability and Systems Engineering, Beihang University.

His research interest includes testing and reliability analysis for software, hardware, and cyber physical system.