Survey on Modeling and Design of Cyber-Physical Systems

NATASHA JARUS and SAHRA SEDIGH SARVESTANI, Missouri University of Science and Technology, United States of America
MARK WOODARD, Sandia National Laboratories, United States of America
KOOSHA MARASHI, Romeo Systems, Inc., United States of America
JING LIN and PRATIK MAHESHWARI, Apple, Inc., United States of America
AYMAN FAZA, Princess Sumaya University of Technology, Jordan

The study of cyber-physical systems (CPSs) is a multidisciplinary research area incorporating elements from statistics, systems management, computer science, and electrical, computer, and civil engineering. Due to their heterogeneous nature and tight coupling between the cyber and physical domains, these complex systems pose difficult problems not seen in traditional infrastructure and computer networks. As such, new modeling and design techniques must be created, and existing methods adapted, to meet these challenges. When modeling and designing complex critical systems such as CPSs, one must ensure they both perform adequately and are capable of providing dependable, safe, and secure service. In this paper, we survey recent literature on modeling and design of CPSs with focus on the critical attributes of dependability, safety, and, security. Literature related to each of these non-functional attributes is introduced, as are techniques for CPS modeling, design, and management that aim to achieve these critical attributes. The research presented in this survey comprises both studies of a general nature, with contributions applicable to a variety of CPS domains, and case studies from a number of specific domains, including smart grids, water distribution networks, and intelligent transportation systems. We conclude the survey by summarizing open research questions related to modeling and design of dependable and secure CPSs.

CCS Concepts: • **General and reference** → **Surveys and overviews**; • **Computer systems organization** → **Embedded and cyber-physical systems**; **Dependable and fault-tolerant systems and networks**; • **Computing methodologies** → *Modeling and simulation.*

Additional Key Words and Phrases: Cyber-Physical Systems, Modeling, Critical Infrastructure, Dependability, Reliability, Measurement

ACM Reference Format:
Natasha Jarus, Sahra Sedigh Sarvestani, Mark Woodard, Koosha Marashi, Jing Lin, Pratik Maheshwari, and Ayman Faza. 2021. Survey on Modeling and Design of Cyber-Physical Systems. *ACM Transactions on Cyber-Physical Systems* 1, 1 (September 2021), 39 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Authors' addresses: Natasha Jarus, jarus@mst.edu; Sahra Sedigh Sarvestani, sedighs@mst.edu, Missouri University of Science and Technology, Computer Engineering, Emerson Electric Co. Hall, Rolla, MO, 65409, United States of America; Mark Woodard, mjw6y7@mst.edu, Sandia National Laboratories, 1515 Eubank SE, Albuquerque, NM, 87123, United States of America; Koosha Marashi, km89f@mst.edu, Romeo Systems, Inc., 4380 Ayers Ave, Vernon, CA, 90058, United States of America; Jing Lin, dzkdlj@gmail.com; Pratik Maheshwari, pratikm@apple.com, Apple, Inc., 1 Infinite Loop, Cupertino, CA, 95014, United States of America; Ayman Faza, a.faza@psut.edu.jo, Princess Sumaya University of Technology, Electrical Engineering, Amman 11941, Al-Jubaiha, Jordan.

XXXX-XXXX/2021/9-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Contents

1 DEFINITIONS AND CHALLENGES

1.1 What Is a Cyber-Physical System?

The rise of large-scale cyber-physical systems (CPSs) has prompted the development of modeling and design techniques that encompass the behavior of both physical and cyber components. This paper surveys a number of these techniques and provides case studies in many application domains. To guide our discussion, we must first understand the defining features of cyber-physical systems and the challenges that cyber-physical system modeling and design must rise to meet.

A multitude of descriptions of ‘cyber-physical systems’ have been proposed. We find that most definitions focus on two system aspects: scale and integration between cyber and physical components. Wang et al. [212] define CPSs as large-scale interconnected systems of heterogeneous components that integrate computation with physical processes. Shi et al. [189] articulate several features CPSs must have, including large-scale networks and tight integration between cyber and physical elements. They must be dependable and able to adapt to the environment dynamically. According to NIST [162], the characteristics of a CPS must include cyber, engineered, and human elements, control loops, and scalable networking.

Wolf [213] argues that a CPS can mean the traditional computer-controlled machine, but large control systems that stretch across an entire country, such as one controlling a power grid, are the end goal of CPSs.

Lee [119] states that CPSs are the integration of computation and physical processes, while Tan et al. [197] define a CPS as a collection of cyber systems that collaborate over a network to monitor and control physical systems. Macana et al. [139] state that CPSs are a combination of components from the cyber and physical worlds where the cyber components monitor and control the physical. Bonakdarpour [24] considers CPSs to be the integration of embedded real-time systems with large collections of sensors and actuators in hostile physical environments. Since CPSs evolved from the field of real-time and embedded systems, Lee [120], Wood and Stankovic [215] argue that embedded systems and wireless sensor networks (WSNs) can in some cases be considered CPSs. Both CPSs and some WSNs have a computational component that includes both hardware and software, and a physical component that is being controlled by an embedded system or WSN.

Throughout this paper we will implicitly define CPSs by studying their applications, challenges, and important attributes. However, we can also provide our explicit definition, which incorporates elements from the above definitions.

A CPS is a system that has two primary subsystems: a physical infrastructure that performs certain physical processes, and a cyber infrastructure which enhances some attribute of the operation of the physical process using computing, communication, and control. The level of coupling between the two infrastructures can vary from system to system, and the scale of the system can vary according to the application, but a reasonable degree of scale and complexity is required to form a CPS. We take the view that tight integration of the cyber and physical infrastructures differentiates CPSs from real-time embedded systems and sensor-actuator networks. Furthermore, wireless sensor networks can only be considered CPSs when the sensing is coupled with communication and control.

1.2 Key Challenges

The large-scale and tightly-coupled nature of CPSs presents many challenges that must be taken into consideration when developing a CPS. In this section, we enumerate the main challenges set forth thus far in the literature in an effort to guide future research towards addressing those challenges.

1.2.1 Interdependence. Interdependence amongst components and operations is one of the most important CPS design challenges because it is impossible to design each component in isolation [84, 95, 119, 187]. Therefore, all key challenges must be targeted simultaneously when designing a CPS. We classify interdependencies as being among the components of the physical infrastructure, among the cyber infrastructure components, and between the two infrastructures [142, 167].

Methods for addressing design challenges include formal verification methods [29], emulation and simulation techniques [71], analytical modeling [119], and using cross-layer design techniques [212].

1.2.2 Dependability and Fault Tolerance. Dependability and fault tolerance are crucial requirements for any CPS [136, 187, 212]. While incorporating cyber control elements into a physical network can improve the physical network's ability to respond to component failure and degradation, it is challenging to add additional components to a system in a manner that increases overall system dependability [207]. Sophisticated system models are necessary to provide quantitative measures of aspects of CPS dependability. CPSs will not see widespread adoption in certain applications, such as traffic control, automotive safety, and health care, unless they can proven to be reliable and predictable [119].

1.2.3 Security. In order for a CPS to operate properly, it must be designed with security in mind. While adding communication and control to a critical infrastructure system is likely to improve its operation, it also adds points of vulnerability to the system. Malicious exploitation of such vulnerabilities can lead to disastrous consequences, including terrorist attacks [136, 149, 187]. Understanding the security risks and determining the correct action to minimize those risks requires a deep understanding of the semantics of the system operation.

1.2.4 Real-Time Aspects. CPSs must handle sensing, processing, and actuation in real time. The physical infrastructure often imposes some time constraints on the cyber infrastructure, requiring it to respond to changes in the physical phenomena within a fixed time period. In general-purpose software, the time it takes to complete a task measures the performance of the system, but has no effect on its correctness. In a CPS, correctness depends on the system's ability to perform a task within a specified time [49]. While solutions to system timing problems may exist for many subsystems in a CPS, the heterogeneous nature of CPSs requires a unified theory that addresses the challenges imposed by this heterogeneity [212].

1.2.5 Communication. One element of designing CPSs is developing communication protocols that enable application components to communicate predictably. Without real-time data, it is impossible to build cyber infrastructure that is tightly coupled to a physical infrastructure; therefore, communication is an essential element to constructing every CPS. Chipara and Lu [37] describe the challenges in communication, which include support for high data rates, real-time communication, responding to changes in priority, sensor mobility, and reliable data transmission. In addition to these concerns, Fan et al. [61] outline the importance of scaling to large numbers of network nodes and interoperating between devices from different manufacturers. Sha et al. [187] note that the interactions among different communications protocols used in a system must be analyzed to prevent adverse behavior. Finally, communication must be secure and must ensure that peoples' privacy is not violated. All these challenges need to be addressed in any implementation of a CPS, and need to be tailored to the specific requirement of the physical infrastructure under control.

1.3 Structure of This Survey

In this paper, we discuss CPS-specific modeling and design techniques that address these key challenges as outlined in Fig. 1. While functional attributes, such as system performance, are domain-dependent, we can discuss nonfunctional attributes, including dependability and security, in a domain-agnostic fashion. We define measures for many aspects of nonfunctional CPS behavior in section 2, as these guide the development of many modeling and design techniques. Once these attributes have been introduced, we discuss modeling techniques that capture these attributes in complex systems in section 3. In both the modeling and design sections, we discuss generic approaches along with case studies in many application domains. We conclude by identifying open questions and essential research tasks.

2 NON-FUNCTIONAL ATTRIBUTES

Two types of attributes characterize systems: *functional* and *non-functional.* The functional requirements of a CPS describe the operational and performance requirements for the cyber and physical infrastructure of that system. Security, interoperability, and reliability, which drive the design of all CPS infrastructures, are examples of non-functional attributes.

2.1 Dependability

Dependability is the ability of a system to provide a justifiably trustable level of service [10]. It describes the behavior of a system over its lifetime: its ability to deliver services and to avoid and recover from faults. Avizienis et al. [10] provide a taxonomy of dependability aspects which we summarise here. Dependability is a broad concept that encompasses many metrics, including availability, reliability, safety, integrity, and maintainability. Metrics may be *qualitative*, describing principles of system design and behavior, or *quantitative*, providing a means to compare different systems' operation. A particular challenge to CPS dependability analysis is unifying definitions and metrics from the various disciplines involved; Kaitovic et al. [106] does this for cyber dependability and power grid dependability definitions and metrics.

In order to define various metrics, we must first understand the types of system events that may be measured. We take these definitions from the work of Parhami [170]. At the lowest abstraction

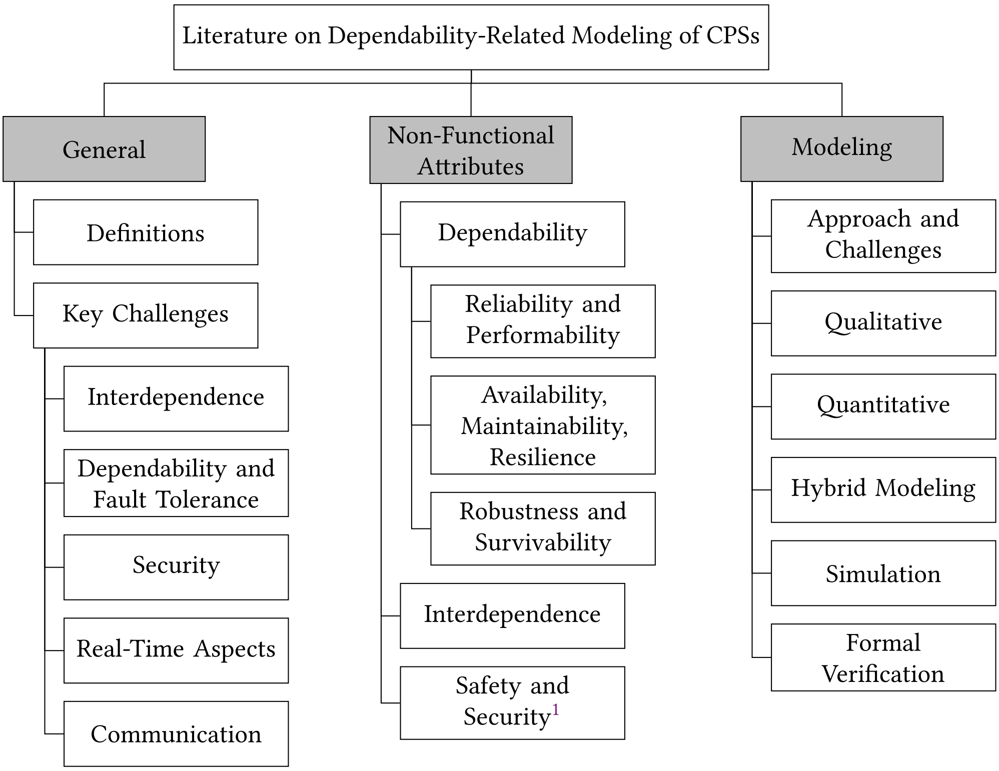

Fig. 1. Organization of This Survey

level is the system component, which may experience a *defect*. A *fault* occurs when a component (whether defective or simply improperly designed) ceases to perform its function perfectly. Faults are undetectable by system monitoring, but may be found through thorough examination. An *error* occurs when one or more faults threaten to compromise system performance. A *failure* occurs when the system is unable to perform as intended; that is, the service the system provides is *degraded*. Failures may be localized to one area of a system (such as a power grid not serving some customers); a *complete failure* causes the system to cease functioning entirely.

System dependability was initially defined in terms of reliability, availability, and robustness. These metrics take a binary view of the system: either it is functional or it has entirely failed. However, these metrics are considered to be too pessimistic to accurately model large-scale systems and thus cyber-physical systems. For example, a nation-wide power grid may experience a service outage in one area, but still be providing service to other areas. This led to the development of more granular metrics, such as performability, resilience, and survivability, that take the level of service a system provides into consideration. Dependability metrics also differ based on which portion of the *system lifecycle* they measure. As such, no one metric can claim to entirely capture system dependability; several models must be used to judge the trustworthiness of a system's service. Figure 2 shows the portions of system lifetime modeled by these metrics.

[1] While both safety and security are important characteristics of CPSes, we focus on dependability-related CPS attributes and direct the reader to other surveys on safety [23] and security [98, 171].

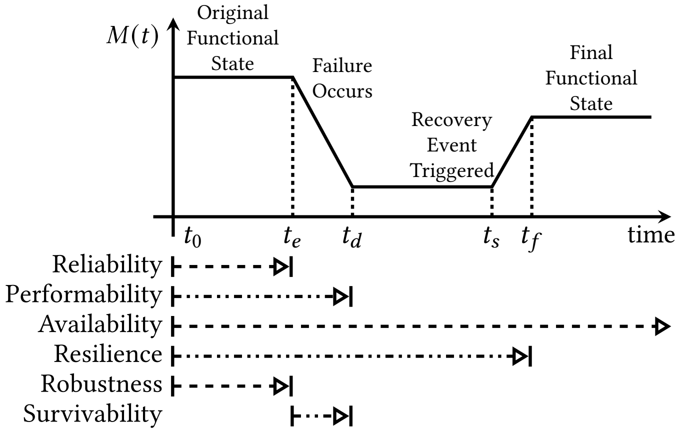

Fig. 2. Dependability metrics and the portions of the system lifecycle they measure. For metrics with dashed lines, $M(t)$ is either Functional or Failed; for metrics with dash-dot lines, $M(t)$ is a continuous measure of system functionality.

Many quantitative dependability evaluation approaches rely on computing a performance metric for a system where some components have failed. Galvan and Agarwal [73] highlights several methodological concerns with this approach. First, simply computing the expected value of the performance metric over all failure cases ignores critical information about the variance and peakedness of the distribution as well as whether certain components have a disproportionate effect on the metric. Second, as the number of possible failure cases combinatorially explodes as the number of failed components increases, the performance metric distribution must be assigned a confidence interval. The authors discuss how this may be done, viewing the uncertainty in the estimated performance metric distribution as stemming from the random sampling of all possible failure cases. Finally, comparing results from real-world systems to results on randomly generated graphs can indicate whether a system's dependability is inherent to any complex system of that size or whether its specific connectivity impedes its dependability.

2.1.1 *Reliability and Performability.*

Reliability and performability are concerned with system behavior before a failure. *Reliability* is the ability of an item to perform a required function under given conditions for a given time interval [202]. However, once the system fails, reliability does not consider partial system functionality or the system's ability to recover; thus, it is a binary measure of continuous operation. In other words, reliability considers every system failure to be a complete failure. Reliability is mathematically modeled using probability. Let X be a continuous random variable representing the system lifetime beginning at the time origin and ending at the instant of system failure. A system's reliability at time t is thus

$$R(t) = Pr\{X > t\}. \tag{1}$$

If we let $F(t)$ denote the cumulative distribution function (CDF) of X, reliability becomes

$$R(t) = 1 - F(t). \tag{2}$$

Or, in other words, the reliability of a system is the probability of it not failing within some interval $[0, t]$. The major challenge of reliability modeling is determining the distribution on X. For small systems, this may be able to be determined through empirical testing, but larger systems require more sophisticated approaches. Often, large systems are divided into small components; the whole system's reliability is then defined as some function of the individual components' reliabilities. Numerous approaches to defining these functions have been proposed, each varying in the exact failure scenarios it can consider, the complexity of systems it can be practically applied to, and its ability to capture system-wide effects such as operator control actions and system maintenance.

One of the earliest reliability analysis tools is the fault tree [124]. Fault trees encode the connections between system components using logic gates. Different types of gates represent the different effects the failure of a component can have on the system. For instance, an AND gate indicates that both subsystems it relates must be functional; this would apply to a system of two components connected in series, among others. An OR gate would model a system with two parallel components that requires both to experience a fault before the system fails. The tree of gates can be analyzed using either analytical or numerical methods [15, 124]. They may also be converted to reliability block diagrams for analysis [141]. While fault trees are intuitive, it is difficult to capture some types of component interdependence with them (but see [153]). Thus, modeling some complex systems with fault trees can be labor-intensive.

Alves et al. [7] apply fault tree modeling to a power grid with traditional generators, wind and solar generators, and battery storage. The supplies each load depends on are determined by exhaustively checking which loads fail when certain supplies are failed. Path connectivity between a load and the supplies it depends on is cast as a k-terminal reliability problem and solved via a DFS algorithm. This information is combined into boolean expressions and incorporated into a fault tree for each load. Load fault trees are then combined into a system fault tree based on which configurations of loads must be functional for the system to be considered functional. The authors use SHARPE [206] to evaluate the fault tree. This formulation does not take into consideration the propagation of failures through a system or the capacities of transmission lines when determining path connectivity.

Reliability Block Diagrams (RBDs) [154] provide another visual means of modeling reliability. Instead of the logic gates of the fault tree, system components are represented as switches in an electrical circuit. If the circuit remains complete between input and output, the system remains functional. As such, RBDs can be used to analyze the probability of the system being functional and thus the system's reliability. Additional analysis can be performed by converting the RBD to a fault tree and applying appropriate analysis techniques [141]. In particular, boolean algebra can be used to reduce the complexity of the RBD and thus simplify analysis [19].

A system reliability model that captures propagated failures is presented in [153]. The authors use a Multi-valued Decision Diagram (MDD), which is a graph-based formalism that represents the logical relationships between multi-valued variables. MDDs resemble logic diagrams, but the gates operate on variables which may take on more than two values. Additional gates, such as functional dependency gates, can capture component dependencies that are not representable using traditional logic gates. The authors propose a new gate to capture propagated failures. This gate represents the causal link between one component failing and that failure propagating to other components. The relationship may be asymmetric; it is possible to capture failure that propagates from one component to another, but not vice versa. The authors propose an algorithm for calculating system reliability based on an MDD and component importance measures based on the reliability analysis.

Markov chains have been applied to reliability modeling in numerous ways [165]. A Markov chain consists of states in which a system may be and transitions among those states that are taken with some probability. The Markov assumption constrains these probabilities: we assume that the probability of transitioning to a given state depends only on the state the system is currently in.

Formally, if we let X_n be a random variable denoting the state of the system at time step n,

$$P\{X_n = x_n | X_{n-1} = x_{n-1}, \ldots, X_1 = x_1\} = P\{X_n = x_n | X_{n-1} = x_{n-1}\}. \tag{3}$$

Kaegi et al. [105] demonstrate how to use agent-based modeling for reliability analysis. A system where each component may be in a functional or failed state is modeled using a Markov chain. Operators that can repair failed components are modeled as agents, which allows the authors to model complex repair strategies and exhibit non-Markovian behavior. de C. Gatti et al. [48] present a method for identifying the agent most critical to operation of the system. This identification is carried out by analyzing interactions among agents. Once identified, services associated critical agents can be duplicated to improve availability and reliability.

One notable technique, the Markov Imbeddable Structure (MIS) method, is presented in [25, 35]. Each state in the Markov chain represents one of the possible permutations of system component failure. The states that result in overall system failure are identified. Transitions between states take place with probability dependent on the reliability of individual components. System reliability is then defined as the likelihood of being in a functional system state after taking one step through the Markov chain for each component in the system. The MIS technique thus models system reliability as a function of individual component reliability.

System reliability can be modeled using the Markov imbeddable structure (MIS) technique [25, 35]. This technique derives a metric for system reliability from component-level reliability measures. Each component can be in either a functional or a failed state. The combinations of component state that result in a failed system are identified. Then, a Markov chain is constructed where transitions between system states occur when components fail. Analysis of the Markov chain reveals the probability that the system remains functional after a certain number of component failures. Faza et al. [64, 66] use this technique to compare changes to cyber control algorithms and cyber device placement in a CPS, as well as the effect of physical faults on the control system. The authors extend their work in [65] by providing methods to reduce model complexity, making the modeling of real-world CPSs feasible with the MIS technique. Marashi et al. [144] apply this technique to compare the effects of cyber and physical component failures on overall system reliability.

Faza et al. [64] develop a method for modeling the reliability of specific power grid topologies; specifically, bipartite power graphs. They consider the effect of the failure of each critical component on system-level functionality; i.e., they enumerate all combinations of component-level failures and determine whether each combination leaves the system functional or failed. This enables calculation of system-level reliability from component-level reliability estimates. The model is applied both to a conventional power grid lacking intelligent control, and to a smart grid fortified with UPFCs. In [65], the authors extend this to more complicated systems, studying methods for reducing model complexity. State space explosion is an obvious impediment to scalability of this model, as 2^n system states result from n components. The authors develop a technique for aggregating components, reducing n by a factor of up to 10 in some cases. Faza et al. [66] utilize simulation and fault injection to investigate how intelligent control can prevent cascading failures in the IEEE-118 bus system. The authors consider various types of UPFC failures, including malicious tampering and control software faults, and determine the effect these failures have on system reliability. Finally, in Faza et al. [63] the authors describe the result of fault injection into the software controlling UPFCs, elucidating how such devices might fail in practice. Their simulation results show that, in certain cases, software failures in UPFCs can cause cascading failure in otherwise functional power grids, underscoring the importance of including software in system reliability models.

Marashi and Sedigh Sarvestani [143] study how control and communication techniques affect the reliability of smart power grids. Using the MIS technique, they derive a probabilistic equation for system reliability that includes transmission lines, FACTS devices, communication paths both wired and wireless, cyber sensors for power quality, control algorithms, and human operators.

Stochastic Petri Nets (SPNs) are another tool used extensively for modeling reliability. A Petri net is a bipartite directed graph where the set of nodes consists of two disjoint sets: places and transitions [140]. Directed arcs connect places to transitions and transitions to places. Each place contains a non-negative number of tokens. The state of the system, referred to as the *marking*, is dictated by the distribution of tokens in various places within the Petri net. The change in the Petri net's marking is controlled the firing condition of the transitions. For instance, a transition may fire once each place that has an arc to that transition contains a token. When a transition fires, tokens are removed from places connected to the transition by an input arc and are added to places connected to the transition by an output arc. The transitions in an SPN utilize exponentially distributed firing times.

SPNs provide a graphical model of system behavior similar to Markov chains. For analysis, they can be reduced to continuous-time Markov chains (CTMCs) to obtain a steady-state representation of reliability [140]. SPNs provide a more concise representation of a system than traditional CTMC modeling, as each marking of the Petri net corresponds to a state in a CTMC. Extensions of SPNs have been proposed for modeling more complex systems, including high-level smart grid control centers [222] and subcomponents such as multi-source power systems [108].

Tien and Der Kiureghian [204] demonstrate an approach to modeling the reliability of large-scale systems with Bayesian networks. A Bayesian network [160] is a quantitative system model based on a directed acyclic graph where each node is a random variable that represents the state of one system component and each edge represents a causal relationship between two components' states. Bayes' Theorem is applied to the graph structure to determine the probability of unknown component states given the knowledge of some component states. For instance, one could compute the probability of system failure based on the failure of a component, or the probability that a certain component has failed given that another component has failed. The authors propose an algorithm for compressing Bayesian networks for large systems by combining interrelated components into clusters that can be modeled by a single random variable. Furthermore, they develop two algorithms for exactly inferring the probability distributions of the random variables in the network.

Zhang et al. [227] present a graph-based model of cyber-physical systems and define a reliability metric based on graph metrics. The nodes of the graph are divided into two disjoint subsets that represent the cyber and physical elements of the system. A node is considered 'functional' provided that it has at least one link to a cyber node and one link to a physical node. A failure is initiated by removing some nodes from the graph. The failure then cascades if removing those nodes cause other nodes to be considered to have failed based on the functionality criterion. The authors define k-reliability to mean that the largest connected component of the graph after a cascading failure contains at least k nodes. Necessarily, this component must contain at least 2 physical nodes and at least 2 cyber nodes. They use this metric to determine the reliability of systems with randomly allocated links and with links allocated following a regular pattern.

Carreras et al. [33] study the impact of heterogeneity on power grid reliability. Most real-world grids follow a so-called "pearls on a string" topology, consisting of homogeneous distribution networks ("pearls") connected by few linking lines ("string"). These linking lines may not have the same reliability as the intra-network lines, leading to heterogeneity in the network; imbalance in power generation and demand inside each distribution network adds another dimension of heterogeneity. The authors generate several grids following this architecture and simulate failures in them. They find that for medium-sized blackouts, having more reliable linking lines improves system resilience, but for large-scale blackouts, the opposite is true as failed linking lines more effectively isolate the blackout. This work highlights the important role that automated switches can play in controlling blackouts.

Ahangar et al. [3] compare the effect of various control network topologies on the resilience of a power grid. The physical system is modeled using a component state-based approach similar to MIS where the probability of each state is determined by the availability of the functioning components and the unavailability of the non-functioning components. The authors use the Expected Energy Not Supplied (EENS) metric to evaluate the reliability of the system. In a case study on a 20 kV power distribution grid in Iran, the authors find a 30% difference in EENS between a power grid where the cyber network is perfect and one where the cyber network may fail. Furthermore, using a mesh control network topology improves EENS 50% over a bus network topology.

Hariri et al. [89] incorporate renewable energy sources and distributed grid storage devices as well as electric vehicle charging demand into their reliability analysis of a smart grid. Due to the complexity of combining the various reliability and energy supply/demand probability distributions, the authors use Monte Carlo simulation to compute the EENS of the IEEE-33 bus system. They find that distributed generation and storage provide a significant increase in the grid's ability to meet voltage and power flow constraints, especially when the grid is cut off from one or more traditional generators. The simulation approach produces results within 0.4% of an exact simulation, but with significantly reduced compute time that allows this technique to scale to much larger systems.

[77] provide a survey of water distribution network (WDN) reliability measures. The authors discuss the various failure modes of WDNs, including pipe failure, pump failure, and reservoir exhasution. Many reliability analyses of pipe failures consider the impact of individual pipe failures. However, depending on the location of shutoff valves, an area larger than the one serviced by the failed pipe may need to be shut down to perform repairs. Thus, some reliability analyses consider segment reliability, rather than individual pipe reliability. The survey also includes a number of performance metrics that cover the behavior of WDNs from several aspects, including efficiency, throughput, cost, and water quality.

The groundwork for analyzing the complete reliability of a water distribution network is laid in [220]. Until this point, reliability analysis had focused on either mechanical failure of network components or the inability of the network to meet demand. First, the performance of various network configurations is determined using a probabilistic hydraulic model. Second, the likelihood of each network configuration is determined and combined with the performance information from the first stage to determine overall system reliability. While the derived solution is only approximate, the modeling technique is applicable to real-world WDNs.

Dasic and Djordjevic [47] develop probabilistic reliability models for WDNs. They consider two perspectives on reliability: the probability of node or pipe failure, and the probability that flow and pressure requirements will be met at every node. Both network-wide and single-node reliability are considered, since single-node reliability can be of interest when that node is important (e.g. a hospital). To calculate resilience, the authors propose several network decomposition techniques, simplifying model evaluation. Finally, they develop a genetic algorithm for designing systems with a fixed reliability while minimizing cost.

Agrawal et al. [2] develop a method for converting an arbitrary WDN topology to one that can withstand the failure of any single pipe without a loss in delivered service to any customer. In their linear programming approach, the objective is to minimize the cost of adding redundancy to the network while maximizing the redundancy gained. The proposed solution relies on iteratively identifying the pipe with the lowest reliability, and either duplicating or increasing the capacity of this pipe. The algorithm terminates when failure of any single pipe does not result in system failure.

Orazio [166] build a reliability model of a WDN that incorporates a model of the hydraulic behavior of the network. This model also incorporates the failure of isolation valves. If an isolation valve fails, significant portions of the network may have to be shut off in order to isolate a failed pipe segment. In small networks, there is an increased chance that shutting off one segment of the network may unintentionally disconnect others; however, larger networks are more likely to have redundant supply paths, reducing this concern for system reliability. The authors develop a pareto-optimal algorithm for planning the location of shut-off valves and demonstrate it in a case study on a WDN in Italy.

Reliability cannot capture the degradation of a system's performance due to failures; therefore, it cannot be used to model non-catastrophic failures where the system maintains partial functionality. Meyer [150] introduces *performability*, which simultaneously captures the performance and reliability of a system. Performability, like reliability, does not consider system behavior after complete failure; it is a measure of system capability before failure. System capability quantifies the extent users can expect to benefit from a system given that it is in a specific state. Iyer et al. [100] discuss developing performability models of fault-tolerant systems that use a capability function, $M(t)$, to relate the state of a system at time t to the overall system performance level.

The performability of a system from the time origin t_0 to time t is given by Equation 4:

$$Perf_{sys}(t) = \int_{t_0}^{t} M(x)\,\mathrm{d}x \tag{4}$$

where $M(t)$ is the reward function associated with performance per unit time and t is the mission time of the system. The mission time of the system is the duration over which the system is expected to be operational. Performability is focused on mission time and becomes difficult to calculate if repairs occur during operation since the mission time can become unbounded. Ciardo et al. [39], Smith et al. [192] present Markov reward models—Markov chains that earn a reward dependent on the state the system is in—for evaluating CPS performability.

Reliability and performability are useful for evaluating systems that are initially in a perfect functional state, but fail to capture repairs after complete failure and thus long-term system operation.

2.1.2 Availability, Maintainability, Resilience. As time goes on, it is inevitable that any complex system will fail. The behavior of a system after such a complete failure is not described by reliability; we must capture this behavior with different attributes. The simplest of these is *availability*: whether or not the system is able to provide correct service at a given time [111]. Like reliability, availability views the system as either completely functional or completely failed.

Effectively, system availability depends on two mutually exclusive events: either the system has not failed before time t, or the system was last repaired at time x and has not failed between time x and t. If we let $m(t)$ be the average number of repairs before time t and $d(t) = \frac{\mathrm{d}m(t)}{\mathrm{d}t}$ be the repair density, the probability of a system being available can be defined as shown in Equation 5.

$$A(t) = R(t) + \int_{t_0}^{t} R(t-x)d(x)\,\mathrm{d}x \tag{5}$$

Goyal et al. [81] develop probabilistic models for steady-state and transient availability, as well as availability over a specified interval. These models are similar to Markovian reliability models, but with added terms for system repair time. They also discuss the difficulties of modeling all types of repair and choosing statistical distributions for repair time. Dyer [54] develops additional

Markovian availability models with more complex state transition rates, as well as a method of approximating model solutions using Poisson processes. Due to its binary view of system state, availability is usually not directly studied within the field of CPSs. Instead, most studies focus on more granular metrics such as system maintainability or resilience.

Maintainability, the ability of a system to be modified or repaired [10], describes the ability of a system to be restored to functionality after a failure. To be maintainable, a system must be designed to undergo preventative maintenance and to easily be repaired after a failure; thus, system maintainability is related to system availability. System maintainability is described by system attributes and operating procedures; as such, it is a qualitative measure of dependability.

Ruiz-Arenas et al. [184] develop nine principles for building maintainable CPSs, drawing on principles from the field of linear complex systems. A maintainable CPS should include fault monitoring that can alert operators when a component requires maintenance. Furthermore, the system must be modeled to predict how it will fail. This information can then be used to both improve the design and increase the effectiveness of preventative maintenance. Preventative maintenance must be scheduled on a regular basis. Finally, when performing maintenance after a failure, operators must work to isolate faults to prevent them from cascading. These principles form a foundation for designing and maintaining highly available CPSs.

Sheu et al. [188] develop a stochastic model of system repair that includes delayed repairs for non-critical failures, immediate repairs for critical failures, and replacement of hardware as it ages. Component degradation and events caused by degraded components are assumed to be nondeterministic; each type of fault has its own probability distribution. The model captures imperfect repair; that is, repaired components may not return to ‘good-as-new’ condition. This technique can be used to determine the effectiveness of maintainability attributes or to judge the resilience of a system.

Resilience takes a more detailed quantitative view of system availability, much as performability does with reliability. Avizienis et al. [10] mention resilience as a synonym for fault tolerance. More specifically, resilience is defined as the ability of the system to bounce back from failure [93]. Ouyang and Dueñas-Osorio [168] expand this definition to include the ability of a system to resist different possible hazards, absorb the initial damage, and recover to normal operation. Mathematically, resilience is defined in Equation 6.

$$\Lambda(t) = \frac{M(t)}{M(t_0)} \tag{6}$$

As with performability, all quantitative resilience measures rely on some measure $M(t)$ of system functionality at time t, alternatively known as a capability function or a figure of merit(FOM) [14, 93].

NIAC [158] qualitatively expands on this definition, emphasizing certain abilities a resilient system must have. Infrastructure resilience includes *absorptive capacity*, the ability of a system to withstand a disruption without impacting service; *adaptive capacity*, the ability of a system to be reconfigured or repaired to meet service requirements; and *recoverability*, the ability of a system to be quickly returned to nominal operation after a disruption. The designers and operators of resilient systems must be able to anticipate disruptive events and plan appropriate action. Avritzer et al. [11] emphasize the importance of considering interdependence between infrastructures when evaluating resilience (see Sec. 2.2). Kwasinski [116] further incorporates a meta-systemic goal of long-term planning, improvement, and adaptation; a resilient system continues to evolve as components age, requirements change, and analysis tools improve.

Ghosh et al. [78] outline a procedure for developing resilience metrics from reliability, performability, or availability models. First, a stochastic model for the chosen dependability attribute is developed. Next, a particular metric on this model is chosen as the measure of system functionality. Finally, structural or parametric changes, which may include component faults, are made to the system, and the resulting change in functionality is observed. Albasrawi et al. [6] apply this methodology to measure not only the loss of functionality resulting from a cascading failure in a power grid, but also the rate at which functionality is regained by different recovery actions. The choice of recovery actions is governed by the maintainability of the system, demonstrating that maintainability and resilience are interrelated.

Nan and Sansavini [157] identify three components of system resilience: ability to absorb faults (i.e., robustness), ability to adapt and reconfigure to reduce the impact of faults, and ability to restore service after degradation. The authors propose a FOM-based resilience metric that captures these components; it also captures the ability of systems to recover to a level of functionality higher than their initial state. They develop a method for applying this resilience metric to complex interdependent systems where no unified figure of merit exists. The interdependent system is divided into subsystems for which a figure of merit can be defined. The interdependencies between subsystems are quantified using input and output variables that allow for simultaneous simulation of subsystem models. Results from simulations can be used to identify the relative effect of each interdependency on overall system resilience and components that can be improved to increase system resilience.

Kwasinski [116] develops an alternative mathematical definition of resilience focused on the ability of a power grid to serve customers. In this model, resilience is given by $R = \frac{\sum_{i=1}^{N} T_{U,i}}{NT}$ where N is the number of loads to serve, $T_{U,i}$ is the amount of time the i-th load is served, and T is the total time of the event being considered. This differs from availability in that T is fixed to the duration of one event and does not encompass long-term system behavior. Other aspects of resilience are also quantified, including the rate of recovery, system resistance to failure, and system brittleness, or the ratio of service outage to system damage. These metrics are applied to data from power outages caused by hurricanes and their use in long term planning demonstrated.

Cybersecurity has independently developed several methods of scoring the vulnerability and resilience of computers; the Common Vulnerability Scoring System (CVSS) is one method based on expert evaluation of discovered vulnerabilities. Jacobs et al. [101] use a control theory approach to relate the performance of a smart power grid and the CVSS Impact Subscore (ISC) approach to quantify cyber-physical resilience. ISC is computed based on the impact of a vulnerability to a cyber system's availability, e.g., a denial-of-service vulnerability that delays communication; integrity, e.g., a signal-jamming vulnerability that adds noise to sensor readings; and confidentiality, e.g., a vulnerability allowing attackers to manipulate sensor values. The authors compute ISC scores for several control system vulnerabilities and compare their effects on control-theory based resilience scores [20]. Such scores measure the ability of the cyber system to control the physical system and the extent of the control inputs required to recover from an attack.

Galvan and Agarwal [72] develop metrics for graph inter- and intra-community centrality with a goal of understanding both the global and local behavior of networked systems. Inter-community centrality reveals nodes essential for that community's operation; intra-community centrality describes the role community nodes play in connecting to other communities. The authors apply these to Britain's railway system, identifying rail lines that carry relatively little traffic but are critical to providing service to certain communities.

Ramirez-Marquez et al. [175] develop a graph-theoretic resilience score based on community detection. They use the similarity between the communities detected in the initial system and the communities detected after recovery from link failures to determine the resilience of a system. The more similar these communities are, the more resilient a system. This approach can also be applied to determine the order in which to recover links based on minimizing the community difference.

Resilience is extended to systems-of-systems by Filippini and Silva [68]. They develop models that incorporate resilience metrics from several systems, including the effects of one system's failure on the resilience of the others. For instance, a blackout in part of the power grid may cut power to water distribution centers and trigger a failure in the water distribution network. They model this failure propagation using a deterministic cause-effect model. However, this may not capture all the effects of other systems on the resilience of the system under consideration, since the state of other systems can affect both the likelihood of a component fault and the maintenance time required to recover the system after a failure.

The resilience of Unified Power Flow Controllers (UPFCs) — a type of Flexible AC Transmission System (FACTS) which consists of several subsystems that control and regulate the power on a line — is investigated by Aminifar et al. [8]. The authors propose a resilience model for each subsystem, then compose the models into a resilience model for the complete UPFC. They also present techniques to reduce the model complexity.

Albasrawi et al. [6] use PSAT and the MIS technique to measure the reliability of a power grid with and without a Static Synchronous Series Compensator (SSSC) FACTS. The authors also simulate and measure the resilience of this grid against particular three-line failures. Their work motivates the challenge of designing smart grids that are both reliable and able to quickly recover from failures when they occur.

Kelly et al. [109] consider how electric vehicles may impact the dependability of the power grid. Such cars would significantly increase load on the power grid, leading to decreased stability. The authors stochastically model the effect of different charging schedules on the sustainability, stability, and resilience of the system.

Hosseini and Parvania [94] model the resilience of smart power grids in hurricane conditions. The smart grid is capable of some amount of automated fault location; several fault isolation switching algorithms are evaluated for resilience. Several metrics for resilience are used, including maximum load loss, load restored by automation, and recovery rate. The authors compare the effect of isolation algorithms and the level of automation present in the grid on resilience, finding that even incomplete automation can greatly improve the resilience of a smart grid in extreme conditions.

A qualitative risk assessment of WDNs is given in [87]. The authors identify terrorist threats and natural hazards that pose a risk to such networks. In addition, they outline 15 different avenues for WDNs to be hardened against such risk. Each hardening approach must consider how the security, redundancy, resilience, and robustness of the network will be affected by a particular policy. Finally, they delineate research areas that provide a foundation for improved hardening approaches. These research areas include quantitative risk analysis, economic or game-theoretic models of attacks, and detailed system case studies.

A qualitative framework for assessing vulnerability of complex infrastructure in the face of multiple threats, the Critical Infrastructure Elements Resilience Analysis (CIERA) is presented by Rehak et al. [179]. This framework incorporates both technical and organizational elements of resilience which are difficult to incorporate in quantitative models. The process begins by identifying and describing system elements and threats to the system. These elements may be system components, subsystems, or even organizational elements such as a security team. Element resilience is scored based on its judged robustness, recoverability, and adaptability, each of which include both technical characteristics and organizational preparedness. Finally, the results are assessed and weak points are identified and improved. The process is applied to a power grid control room; this case study finds that the weakest aspect is the organization's resilience due to its rigid structure and low investment in mitigating cyber vulnerabilities.

2.1.3 Robustness and Survivability. In addition to being maintainable, dependable CPSs must be designed to be robust in order to meet the demand for uninterrupted service in the face of component failures. Siewiorek et al. [190] define system robustness generally as the ability of a system to tolerate errors. NIST [161] defines robustness as the ability of a system to operate correctly and reliably across a wide range of operational conditions. Rungger and Tabuada [185] define robustness for CPS as input-output dynamical stability, where bounded inputs have bounded effects on the system. In the CPS domain, robustness is defined as the ability of a system to tolerate errors without a reduction in performance. Žiha [229] delineates the difference between redundancy and robustness: robustness is the ability of the system to respond to all possible failures, whereas redundancy is concerned with local component failure. A robust system should be redundant, but a redundant system may not be robust.

Koç et al. [113] provide a quantitative measure of robustness as the ratio of errors that cause a reduction in performance to the number of possible errors. For systems with continuous states, such as power grids, robustness is closely related to the excess demand each element of the system can sustain without failure. In the systems studied by the authors, robustness could be accomplished using redundancy.

Rao et al. [176] present a graph model for CPSs and methods to compute the robustness level of the system. They focus on making both the cyber and physical portions of a networked infrastructure robust by ensuring the system meets performance requirements with a specified probability in the presence of cyber and physical degradations due to natural, accidental, and intentional errors. They construct a cyber-physical network infrastructure graph model of the system's components. Game theory techniques are then used to model strategies attackers may use to compromise the system. The result is a measure of how robust the system is to both natural and human-caused failures. Yagan et al. [221] model a CPS as a cyber network overlaid onto a physical network and evaluate system robustness based on the allocation of interconnected links between nodes in the two networks. They also present an algorithm for allocating cyber-physical interconnection links in CPSs. The algorithm is optimal against random attacks on CPS networks with unknown cyber and physical topology.

Koç et al. [115] produce a robustness metric that captures system topology, power flow, and node significance. Topology and power flow are combined using an equation that draws inspiration from information-theoretic entropy computation. The authors use this metric to determine the effect of adding lines to a power grid on system robustness. Critically, this approach is not simulation based and thus is deterministic and scales to very large systems.

While robustness captures the ability of a system to tolerate an error, it does not capture a system's response to errors that cause a degradation in performance. Once the system's performance degrades, its analysis enters the purview of *survivability.* Robustness measures the ability of a system to avoid failures; survivability measures a system's ability to remain dependent after failures occur.

The roots of survivability are in military applications which focus on mission fulfillment. Most definitions of survivability are qualitative; for example, Ellison et al. [57] define it as the capability of a system to fulfill its mission in a timely manner in the presence of attacks, failures, or accidents. The mission of a system is a set of very high-level requirements or goals for that system; timeliness means the mission is fulfilled by a user-specified time.

Queiroz et al. [174] define survivability as the capacity of essential services to provide their functionalities in cases of malicious attacks compromising parts of the system. Such functionalities may rely on other services of the system which are not necessarily essential. The definition focuses on a specific service or component that must survive and how the interdependency between services affects that survivability.

There is no NIST definition for survivability; however, a number of NIST definitions for other domains cover elements of survivability. The NIST definitions of robustness and resilience for Information Assurance include the ability of a system to fail gracefully and operate in a degraded state while maintaining essential capabilities [161, 181]. These two definitions describe survivable behavior of a system.

Attacks, failures, and accidents are included in the definitions because they are all potentially damaging events. The definitions focus on mission fulfillment, not on specific subsystems or components that must survive. From them, we can distill a number of characteristics of survivable systems:

Resistance to Attacks: Systems will be designed to repel attacks using strategies such as user authentication or stochastically diverse programs.

Recognition of Attacks: Systems will have mechanisms to detect attacks and determine the extent of damage to understand the state of the system after an attack. Possible strategies include intrusion detection and internal integrity checking.

Recovery: Systems will be designed to recover essential services after an attack and to be able to return to full capability. This includes restoring compromised information as well as functionality within the time constraint dictated by the mission. Strategies include data replication, backup restoration and system reinitialization.

Adaptation and Evolution: Systems will be designed to adapt and evolve to reduce the effectiveness of future attacks. One strategy to achieve this characteristic is to improve intrusion detection using knowledge gained from previous attacks.

Some of these characteristics fall under other non-functional attributes of CPSs. CPS survivability involves graceful degradation, limiting fault-propagation, and minimizing effects on interdependent systems.

Very few survivability definitions are mathematically precise enough to be used with models and simulation to determine if system survivability requirements are met [112]. The T1A1.2 working group, however, does provide one quantitative definition of survivability [83]:

> "Suppose a measure of interest M has the value m_0 just before a failure occurs. The survivability behavior can be depicted by the following attributes: m_a is the value of M just after the failure occurs; m_u is the maximum difference between the value of M and m_a after the failure; m_r is the restored value of M after some time t_r; and t_R is the time for the system to restore the value of m_0."

This definition of survivability captures the qualitative description of survivable behavior as well as the time-varying behavior of the system after a failure occurs.

A number of approaches have been used to model survivability. Zhang et al. [226] present a qualitative approach using attack graphs. In this approach, an attack graph is created using known system vulnerabilities and their associated difficulty parameters. Each path represents a series of exploits leading to an undesirable state. Each node represents the network states under attack and each directed edge represents an attack action. Survivability analysis is conducted by defining the states in the attack graph where the system fails completely and determining the cost associated with each attack. In this case, survivability is associated with the difficulty and the destruction level of an attack, quantitatively defining survivability as the minimal cost to compromise the system. This model captures the probability that a system will meet its mission requirements in the presence of an attack, but neglects the presence of survivable system enhancements. It does not model the timeliness or ability of a system to recover or the graceful degradation of a system.

Liu and Trivedi [135] introduce a method of modeling survivability using continuous time Markov chains (CTMC) by combining a pure performance model and a pure availability model to construct a composite performance–availability model. The pure availability model for a system's resources is modeled as a birth-death process where each state represents the number of functioning assets. The pure performance model is created using task arrival rates and service rates for the system. It is a birth-death process, where each state represents the number of assets currently tasked. The two models are combined to create a composite model which is then truncated based on the survivability measure of interest. The desired survivability measures are obtained using transient analysis. Many extensions to this model have been presented to incorporate different aspects of survivability. Cloth and Haverkort [42] developed a checking algorithm to decide whether a system is survivable. Continuous stochastic logic [12] is used to phrase survivability in a precise manner for CTMC models. Heegaard and Trivedi [92] expand and refine this method to determine the scalability of the model as well as to model additional performance measures, such as failure propagation and recovery using a phased recovery model.

Kim et al. [110] model the survivability of a wireless sensor network using a Semi-Markov Process (SMP) instead of a simple Markov chain. The SMP captures that because the behaviors of attacks, system responses to the attacks, intrusion detection, and repairing mechanism cause sojourn time to be non-exponential.

System survivability has also been modeled using Petri nets. Castet and Saleh [34] explore the applicability of stochastic Petri nets for multi-state failures and survivability analysis. They model components with multiple operational states, allowing them to analyze survivability and focus on failure propagation in the system that results in either graceful degradation or catastrophic failure.

Ghasemieh et al. [75] use a Hybrid Petri nets model to evaluate the survivability of fluid systems, specifically, the survivability of a waste water treatment facility in the presence of component failures or bad weather. Using this model they are able to determine under which circumstances the system will overflow and thus fail.

Woodard et al. [216] use this simulator to demonstrate a simulation-based approach for evaluating the survivability of networked systems with an arbitrary known topology and provides a technique for finding susceptible critical components based on this evaluation.

The effect of attacks on or failures in a WDN can be mitigated by isolating, i.e., temporarily withholding service to a number of nodes. Selecting these nodes is a complex problem. Jeong and Abraham [102] develop a genetic algorithm that generates strategies that attempt to minimize the degree to which critical services are disrupted, the economic impact of the failure, and the number of people affected. Their algorithm generates multiple strategies that provide different optimal solutions to the service denial problem. Policy makers may then choose the strategy that best fits the exact situation.

2.2 Interdependence

Rinaldi et al. [180] provide a qualitative analysis of interdependencies among the electric, water, gas, oil, and telecommunication networks. The authors describe how a failure in one of the systems, such as the power grid, can cause disruptions in other systems, such as curtailment in the production of natural gas, or disruptions in irrigation pumps in the water distribution system. Laprie et al. [118] introduce qualitative models for analysis of the interdependencies between electricity and information infrastructures. The study addresses three types of failures that are of particular interest in interdependent infrastructures; namely, cascading failures, escalating failures, and common-cause failures. In [53], a graph-theoretic approach is used to model interdependencies among four networks; the power grid, the gas network, the water delivery system, and the transportation network. The study analyzes network resilience, which is defined as the capacity to remain connected after vertex removal, and discusses fragmentation of networks due to failures. Eusgeld et al. [59] discuss a system-of-systems approach for modeling interdependence among subsystems in a CPS. After comparing a number of related methods, they conclude that agent-based modeling, high-level architecture, and hybrid systems show the most promise.

Vatn et al. [210] categorize system interdependence based on risk, vulnerability, and criticality of the system being studied. Risk vulnerability assessment is used in [209] to infer the effects of interdependence from observed failures. The authors provide a hierarchy of analysis steps that progressively reveal greater detail about infrastructure while analyzing the failure event.

Marashi et al. [142] present an approach for quantifying and analyzing interdependencies in CPSs. The authors identify sequences of cascading failures and build a graph of dependencies among components based on these sequences. From these dependencies, dependency indices are calculated that rank physical-physical, physical-cyber, cyber-physical, and cyber-cyber dependencies in order of criticality.

Huang et. al. [96] model the interdependence of the power grid and communication/control networks utilized in the Smart Grid. A failure in one of these networks can cause failures in the other potentially leading to cascading failures. They calculate the size of the functioning area of the network after a cascading failure using percolation theory. From this they determine the survival ratio of functioning to non-functioning nodes. Their analysis reveals a nonlinear relationship between the number of controlling nodes and system survivability. They determine through both mathematical and experimental results that a smart grid with more controlling nodes is more survivable.

Beccuti et al. [17] present a quantitative model that captures the interdependence between the physical and cyber realms of a smart power grid. They use a stochastic well-formed net for detailed representation of system protocols and avenues for denial-of-service attacks, facilitating analysis of the effect of these attacks. The model is used to determine the effect of a denial-of-service attack on the power grid as cyber control actions are taken.

Verma et al. [211] compare the sensitivity of graph-theoretic importance metrics (betweeness, closeness, and degree centrality) to that of a power-flow-aware node significance metric. The sensitivity of each metric is tested by removing the node considered most important by that metric and computing the power flow of the resulting system. For completeness, the metrics are also compared against a random node removal strategy. The authors' results demonstrate that removing the node considered most important by the node significance metric results in a power grid with much lower performance than any other metric. Notably, the graph-theoretic metrics all perform on par with random node removal. The authors conclude that node importance metrics that negelect the dynamics of the physical system fail to capture essential information about that system's behavior.

Ezell et al. [60] outline an Infrastructure Risk Analysis Model to determine how the interdependencies and interconnectedness of a WDN affect its safety. They define four steps: identifying risks, modeling the risks, assessing the infrastructure's ability to withstand damage, and managing risk to the infrastructure. This methodology provides a holistic set of guidelines to policy makers, showing the tradeoffs between cost and safety.

Katina et al. [107] discuss how interdependence affects healthcare infrastructure. Beyond simply studying the hardware and software of the infrastructure and providing guidelines or metrics to improving it, the authors consider the role that operators and policy makers play in interdependence.

Huang et al. [97] takes a graph-theoretic approach to modeling interdependence between a power grid and its control system. As the controls require power, they implicitly depend on the infrastructure they control. The authors model this as a directed bipartite graph of cyber and physical nodes where the edges represent dependencies. Each cyber node has an edge to the physical node that powers it; each physical node has a bidirectional dependency on the cyber node that controls it. Edges from physical nodes to physical nodes or cyber nodes to cyber nodes are allocated following a scale-free graph model. A failure begins in one part of the bipartite network; it is then propagated as failed components cause their dependents to fail. The authors observe that increased connectivity increases the number of surviving nodes, that is, the number of nodes in the giant component of each part. The effect is asymmetric between the parts; increased cyber connectivity has a stronger effect on the overall system resilience.

Banerjee et al. [13] model one-step interdependencies in a system as where each component's dependencies are expressed as a statement using logical conjunction and disjunction. This model formalism allows for complex relationships among components to be specified. In each time step of a simulation, components whose sentence evaluates to false are marked failed; the complete extent of failures is thus a fixed point of this iterative process. While the formulation of these sentences is challenging for complex systems, they bring several benefits. The authors apply them to produce optimal sets of components to harden in order to limit failures to a specific amount or to keep costs within a specific budget. They prove that this problem is NP-complete, provide bounds on how difficult it is to approximate solutions, and present heuristic algorithms for generating solutions.

3 MODELING

Developing accurate models of CPSs is essential to analyzing and improving CPS architectures. This section will first discuss modeling challenges and general qualitative and quantitative methods for modeling CPSs. Different modeling techniques will then be examined, along with solutions to some modeling issues. Finally, we will present case studies from various CPS domains.

3.1 General Approach and Challenges

All CPS models must capture both cyber and physical components and behavior of the system. A CPS model is comprised of models of physical processes, control software, computation platforms, and networks [49]. CPS models fall into two broad categories: *qualitative* and *quantitative*. Qualitative modeling takes a high-level view of the system, describing system components and their interactions. Detailed mathematical models of specific physical and cyber system components are the realm of quantitative modeling. Qualitative models frequently drive the development of quantitative models and provide a framework for combining disparate quantitative models into a unified system model.

Representing both cyber and physical attributes in a single model is a significant challenge. In CPS modeling, the sensitivity of physical systems to time delays must be captured, along with the delay the cyber control system experiences between sensing and reacting to the physical world [49, 197]. Timing is critical to the correctness, as well as the performance, of the control system. Along with timing delays, system concurrency must be modeled; every real-world CPS will feature multiple control units acting simultaneously on different parts of the physical infrastructure [88].

Most real-world CPSs depend not only on their own physical and cyber elements, but also on other CPSs. For instance, the cyber portion of a power grid CPS depends on the telecommunications network, and the physical hardware running the telecommunications network depends on the power grid. As such, failures in one CPS can cause cascading failures in other CPSs, and a vulnerability in one CPS can be exploited to cause failure in another. Interdependence modeling is two-pronged: single-CPS models must capture dependencies among physical and cyber components of a system [68, 97, 142], and multi-CPS models must capture all of the possible dependencies among the CPSs they model [85, 86, 148].

Implementing software to model CPSs necessarily involves elements of software engineering. Abstract modeling languages such as Unified Modeling Language (UML) can be used to create models from a high-level perspective. Code generation tools can be applied to these models to generate both simulation software and cyber control software used in the CPS itself. Model composition allows detailed models of elements, such as sensor nodes, to be composed into a larger model that represents the complete CPS. Software engineering patterns, such as aspect-oriented programming (AOP), provide useful abstractions and frameworks that ease the complexity of implementing modeling software. Software quality and correctness tools can be used to validate models and prevent simple mistakes, such as mismatches between units in equations [49, 196, 214].

Models play several roles in CPS development: analysis, synthesis, verification, and the oft-overlooked role of communicating design features to other humans [122]. Broadly speaking, modelers take either a prescriptive (termed ‘engineering’ by Lee et al.) or a descriptive (‘scientific’) perspective when constructing a model. System specifications are one kind of prescriptive model: a description of how a system ought to behave. With a prescriptive model, the model is created first and the designers’ goal is to construct a system which follows the model. By contrast, a descriptive model starts with a system and describes how it is behaving. In CPS design, descriptive models are often made to check that a system meets its specification; this process is known as model verification. Both perspectives thus play a critical role in system design.

CPS design requires multiple model formalisms at different levels of detail; one way of viewing the relationship between these levels of detail is in terms of abstraction and refinement [122]. A model is a sound abstraction of another if it preserves “interesting” properties of that model; a model refines another by adding additional detail without contradicting it. Several modeling techniques can be cast in these terms. Analyzing a model consists of making a more abstract model that preserves whatever property is to be analyzed. Model verification checks that some model matches a specification — a more abstract model. By contrast, a physical system is not a model and thus not a refinement of a specification. Instead, a model is faithful if properties of the model also hold for the corresponding system. Validation is then the process of checking that a model is faithful.

Lee [121] identify several limitations of CPS modeling. First, many models, including models of real-world systems [203], exhibit chaotic behavior: small perturbations in inputs lead to vastly different outcomes. Second, incorporating both discrete and continuous behavior in a single model introduces several modeling issues. Causal loops can occur where a model’s input at time t depends on its output at time t and vice versa. Models can exhibit Zeno behavior, where an infinite quantity of events occur in a finite time, such as in a simplistic model of a ball bouncing on a rigid surface. Furthermore, even detecting Zeno behavior is difficult: depending on the power of the modeling paradigm used, it may be as difficult as proving theorems, making a general algorithm for detecting Zenoness intractable. Finally, defining model formalisms that only produce deterministic models may also be intractable. The authors construct a model where two events happen at a particular time offset t; for all $t > 0$, the model is deterministic, but at $t = 0$ the result depends on the order in which the events are considered. These limitations show that modeling is an inherently complex field; when constructing models and interpreting results, modelers must keep in mind the limitations of their modeling formalism and understand the results their analysis approach produces.

3.2 Qualitative Modeling

Precisely capturing the complexities of CPSs is not an easy task due to the heterogeneity and connectedness of components in the system. Models of a single CPS focus on abstracting the operation of the components into modules that are assembled into an overarching model. In Ilic et al. [99], Xie and Ilic [218], a CPS is viewed as a set of non-uniform modules connected through both cyber and power networks. Not all physical components can be modeled from first principles due to their complex behavior. These devices are instead modeled as a cyber-physical module consisting of a physical device and a cyber sensing and control device. Module dynamics can then be discretized based on the sampling rate of the sensing device. Interdependencies among modules are captured in the model, which serves as a basis for studying cyber improvements to the modeled CPS. The combined model is used to determine the dynamics of the entire system and the effect of various control actions on those dynamics. Thus, system performance can be analyzed in spite of the complex interdependencies among modules.

Instead of considering a modular approach, Lin et al. [127, 129] model the relationship between the cyber and physical using an ontology. Each cyber component is an actor that takes sensor data, interprets it using the ontology, and decides on an action. This ontology captures the behavior and relationships among components, whether cyber or physical. Fitch et al. [69] also take an ontological approach to CPS modeling, incorporating critical systems heuristics from the social sciences. The proposed model captures not only hardware and software components of a CPS, but also human factors.

The future of autonomous vehicles in transportation systems presents additional challenges. Koopman and Wagner [114] discuss the challenge of creating safe ITSs. The need for extremely high safety levels requires system validation and verification approaches for both normal and abnormal environments and in the presence of system faults and partial failures. To address some of these challenges, a systematic UML model-based method is proposed in [18] for planning validation and verification.

3.3 Quantitative Modeling

In contrast to qualitative modeling, quantitative approaches focus on developing mathematical models of systems and system components.

A number of studies focus on the interaction between cyber-enabled components of a system and the central control system that governs those components. Ravindran and Rabby [178] divide a CPS into an Intelligent Physical World (IPW) and an Intelligent Computational World (ICW). The IPW consists of the physical world and locally-acting cyber modules. It has three essential traits:

Programmability: The ability to specify how the ICW interacts with the physical world.
Observability: The ability for the ICW to know the inputs and outputs of the IPW.
Computability: The ability for the cyber modules in the IPW to generate outputs to the physical world at a reasonable rate.

The ICW is comprised of the network connecting the cyber modules and the system providing global control over the IPW. It is characterized by its ability to:

- generalize control across diverse networks and IPW devices,
- uncertainty in its model of the physical world, and
- be resilient to network failures.

Separating the ICW and IPW in this fashion allows for simpler mathematical models of the control system.

A different approach to control network modeling is taken in [222], where the control network is modeled using a stochastic Petri net (SPN). The SPN provides a state-based model of the system from which transient and steady-state behavior can be determined. System reliability and availability with no backup, cold backup, and hot backup are calculated. The authors also provide insights on reducing the size of the SPN state space, which is prone to explosion when modeling real-world control networks.

Monte Carlo simulation is a powerful tool, but on models with large numbers of parameters, it becomes computationally infeasible to generate results with high levels of confidence. This is due to the need for numerous simulation runs, varying parameters on each run. Schupfer et al. [186] address this problem with range-based system simulation. They propose a semi-symbolic solution using affine arithmetic that models the system numerically, but provides symbolic ranges for system deviations. The result is a single solution that models the system output for the specified range of system input.

FARE, a package for failure analysis and reliability evaluation for generic CPSs, is developed by Wu and Kaiser [217]. It aims to provide benchmarks for design-time analysis and continuous runtime metrics for interdependent CPSs. The tool incorporates a wide variety of analysis techniques for both failure detection and reliability evaluation.

Andrijcic and Haimes [9] use metamodeling techniques to unify engineering, social, and economic perspectives of bridge maintenance. Each domain is modeled individually as an optimization problem involving several state variables and constraints. The models are then combined by defining functions which map some of each model's state variables to a shared state variable: bridge traffic capacity. Shared state is identified by constructing system dynamics diagrams for each perspective and identifying common influences. This technique is applicable to metamodeling problems where each perspective can be captured using the same formalism.

Another multimodeling technique based on model refinement is presented in [16]. A railway heater system is studied where heater controllers request power from a central distribution center. Each controller, as well as the central control unit, is modeled as a contract automata. These models are composed together; however, the resulting automata permits undesired behaviors, such as a heater not being powered even when there is energy available for it. The authors define a notion of refinement that allows them to eliminate these states without introducing out-of-spec behavior. These models are then mapped to stochastic activity networks which allow the specification of continuous-time dynamics, including rail thermodynamics and weather patterns. The result is a metamodeling system where discrete and continuous dynamics can be correctly related.

A number of ITS models have been proposed to assist in the design of control systems. These models target different elements in the systems. The macroscopic continuous Petri net traffic model proposed in [104] captures various traffic modes such as free-flow traffic, traffic jams, and stop-and-go waves. This model can be used to create a predictive control strategy. Another traffic model, proposed by Fanti et al. [62], aims to estimate the state and control of freeways. This model uses first-order hybrid Petri nets to capture the continuous and discrete behaviors of traffic systems. This model becomes the basis of a networked control strategy for coordinating speed limits to maximize flow. The model proposed in [50] focuses on control of real-time urban traffic lights. A colored Petri net model is used with a genetic algorithm to optimize control of traffic light. This model uses vehicle-to-infrastructure communication to obtain data. Grether et al. [82] develop an agent-based simulation of the interaction between vehicle agents and transportation systems. This simulation utilizes a queuing model to capture traffic flow and spill-back at congested intersections. Similar models have been proposed for rail and aviation systems [30, 228].

When constructing dependability or performance metrics for systems, it is necessary to validate them on a variety of systems. Instead of constructing several large power grids by hand, Thacker et al. [201] develop a power grid synthesis algorithm that replicates key properties of existing power grids. Their algorithm generates random power grids with the same hierarchical structure and node (generator, transformer, bus, or load) degree distribution. It also replicates physical attributes: node location and edge (power line) length distributions, both of which play a critical role in power grid performance. This algorithm can also be used to generate detailed power grids from, say, the high-voltage portion of an existing network.

3.4 Hybrid Modeling

Differences in time scales pose a significant challenge to design, modeling, and simulation of CPSs [119].

Quantitative CPS models can be classified based on their representation of time — a significant challenge in representing both cyber and physical elements in a single model. Control software must decide how continuous-time physical events are encoded into a discrete-time system, and CPS models must incorporate both continuous-time and discrete-time system behavior. Tan et al. [198] develop a CPS model that treats all events as if they have discrete time. The authors argue that since the control system perceives time as discrete, the continuous nature of the physical world is invisible to the control software and thus the model. Events, which include both observations of the physical world and control actions, can be either instantaneous or interval-based, and can be related using temporal operators such as Before, After, During, Begin, and End. The model provides means for tracking where in the CPS events are generated and processed.

The continuous nature of the physical world is addressed by Rovers et al. [182], who develop CPS models that mix discrete and continuous time. Physical signals are stored as continuous mathematical functions that are evaluated at discrete times as needed by the cyber modules. Sensor nodes generate these physical signal functions and intermediate processing nodes are modeled as functions that operate on the input parameters to signal functions. Riemann sums of the resulting functions allow the model to capture system behavior over periods of time. Fitzgerald et al. [70] demonstrate an alternate approach to combining continuous and discrete event models. The models are evaluated through simulation; time is synchronized between simulators for each model of the system. This process of co-simulation requires that each simulator is capable of communicating values to and from the others; continuous-time simulators must be able to inject discontinuous discrete events, and discrete-time simulators must be able to sample the continuous model values at some frequency. Further development of timing models is an open research area.

An application of state-based temporal semantics is the PTIDES project by Eidson et al. [56]. A network-wide time server is used to synchronize many control systems, providing a global standard for temporal semantics in a distributed system. The system is capable of coordinating simultaneous actions across multiple systems without delay that could otherwise have severe effects on the physical world. Cardoso et al. [32] investigate the limitations of network time synchronization, including network latency and variance in network latency. They put forth suggestions for improved network hardware and software, as well as providing a foundation for modeling network delay in CPSs.

The Architecture Analysis and Design Language (AADL) [67], a set of formal modeling concepts for describing complex systems, forms a foundation for several CPS modeling tools. ADAPT [183] is a tool for generating stochastic Petri net models of system dependability from AADL architectural models. Hecht et al. [91] extend AADL and ADAPT to produce stochastic analysis networks for modeling the reliability of systems. AADL is extended with temporal semantics in [88]. The authors introduce a durational calculus that encodes timing information on system states and state transitions. System properties such as safety, liveness, and reliability are formulated in the calculus, allowing for basic formal reasoning about system properties with respect to time. Basing these modeling tools on AADL allows each of them to be easily applied to the same system architecture representation.

Bradley and Atkins [26] propose a system for co-modeling cyber and physical infrastructure. This combined model allows the cyber infrastructure to understand the effect of both physical actions on the physical world and cyber actions (such as changing sensor sampling rates) on the control system. The complete CPS is modeled as a continuous linear system. Solutions to this system optimally balance physical system stability and cyber computational load, which allows for system designs that are flexible with respect to usage of power and computational resource usage.

Ten et al. [199] focus on detecting anomalies in the Supervisory Control and Data Acquisition (SCADA) controls of a power grid that may be caused by malicious activities. The authors provide a taxonomy of SCADA information sources. Using this taxonomy, they correlate the temporal and spatial locality of events that occur during normal operation or under common failure modes. If the system detects events that are not correlated, it can report these as likely malicious actions.

Another tool for composing discrete- and continuous-time models, The Modelverse, is presented by [200]. This tool provides several domain-specific languages for specifying state machines, system architecture, temporal predicates, and physical dynamics. The Modelverse has a formalism transformation graph that defines transformations between model types. This is used in tandem with an architecture model to define how the various models are composed into a single Petri net model.

Bliudze et al. [21] identifies several challenges in CPS modeling and design. Two physical modeling languages are presented: linear graphs and bond graphs. Both have trade-offs: linear graphs sometimes force an unusual choice of variables on designers, while bond graphs cannot be easily composed and thus require designers to describe the entire physical system in one model. Once these models have been constructed, they must be discretized for computer simulation. In this process, designers must make assumptions about the system, such as the minimum interval between two discrete events. Physical model results can be very sensitive to initial conditions; furthermore, certain conditions can lead to multiple solutions. Thus, choosing these automatically is difficult. Finally, for systems where various subsystems require different solvers, coordinating time and results between them is challenging.

The issue of representing time in hybrid models is a universal challenge. Cremona et al. [46] identify several issues with the common choice of using a single floating-point number to represent time. Floating-point numbers are not equally spaced; as the magnitude of a number increases, the gap between consecutive numbers becomes larger, causing the precision to decrease. The effect is that the choice of the time origin can affect the precision of simulation results. In addition, this causes addition to not be associative: if t_1, t_2, t_3 have different magnitudes, it is not guaranteed that $(t_1 + t_2) + t_3 = t_1 + (t_2 + t_3)$. This causes equality to become meaningless: if t_2 has a large magnitude, $(t_1 + t_2) - t_2 \neq t_1$. Thus, just because two events have the same floating-point timestamp, we cannot know they occur simultaneously. Furthermore, using a single timestamp does not allow encoding of causality among simultaneous events. To address these issues, the authors propose superdense time, where time is represented as a tuple (t, m) where t, the model time, is a real number and m, the microstep, is an integer. Two events are understood to be simultaneous if the model time of their timestamps agree. The microstep allows simultaneous events to be ordered, preserving causality even if two discrete events occur at the same time. For simulation purposes, they suggest approximating the model time as an integer multiple of some minimum time step. Finally, they provide an extension to the Functional Mockup Interface and define methods for converting these timestamps to and from floating-point timestamps for simulators.

3.5 Simulation

Loki is a tool for modeling fault injection in distributed systems [36]. It instruments part of a distributed system, capturing the system state as visible from each instrumented node. Each instrument can then inject faults and observe how they propagate through the system. The observations are then used to generate measures of system performance in the presence of faults.

Integrating existing software tools, rather than developing brand-new software, is a cost-effective approach to CPS simulation. A tool that integrates the ns-2 network simulator [164] with the Modelica framework [155] is described in [4]. As the Modelica framework is a modeling language for large-scale physical systems, integrating it with ns-2 allows the physical world to be simulated jointly with a controlling actuator network. An update to the work, [5], addresses synchronization between Modelica and ns-2, as asynchronous event handling can have catastrophic effects on system functionality. Pan et al. [169] propose another co-simulation framework that integrates power grid and communication network simulations utilizing MATLAB and ns-3 [163], the successor to ns-2. Chu et al. [38] use ns-2 and OpenDSS [58] to simulate network effects on smart grid performance when dynamic demand response controllers are used.

In [193], the DIgSILENT Power Factory [51], MATLAB Simulink [146] (including the SimEvents toolbox [145]), and the Matrikon OPC Server [147] are integrated into a unified simulator that captures the effect of cyber-attacks on the performance of CPSs. Marashi and Sedigh Sarvestani [143], Woodard et al. [216] combine PSAT [151], a MATLAB-based power systems simulator toolbox, with libraries that simulate cyber infrastructure and form an integrated smart grid simulator. This package simulates the behavior of measurement, control, and communication systems and provides decision support algorithms. Lin et al. [126] combine EPANET [208] and MATLAB to simulate a water distribution network, where the MATLAB program simulates the cyber network and provides inputs to EPANET, which simulates the pipe network.

Brooks et al. [28], Goderis et al. [80], Ptolemaeus [173] present Ptolemy, a tool for multimodeling of CPSs. Ptolemy is very flexible in what it can model; users can create their own model types and develop algorithms for model evaluation. Ptolemy supports integration of heterogeneous models, where component models of different types are integrated into a complete system model.

Clark et al. [40], Courtney et al. [43, 44], Gaonkar et al. [74] develop Möbius, a tool that combines many model solution methods and formalisms into a single package. The infrastructure provided by Möbius abstracts model details, allowing users to develop plugins to integrate different modeling programs. Users can express models in different languages and apply various solvers to them, making it easy to compare solution techniques. Bohnenkamp et al. [22] combine Möbius [40] and the MoDeST modeling language [55] to model both qualitative and quantitative aspects of CPSs. System models can be formally verified with Möbius's model-checking capabilities.

Al-Hammouri et al. [4] have developed a method for simulating a power grid as a sensor-actuator network, using Modelica to model the physical world and ns-2 to simulate the sensor network. Because the network portion both senses and acts upon the physical world, synchronization of the two simulators is required. This is done by allowing ns-2 to stop and start Modelica as needed, preventing Modelica's in-simulation time from exceeding the in-simulation time of ns-2. The end result is a simulation of a cyber control loop that regulates the voltage of a power supply for a variable load, taking into account network delays and losses that prevent the system from instantaneously reacting to varying system load.

Instead of looking at the power grid as a single complex system, Xie and Ilic [219] envision it as an interconnected series of modules. Each module has defined goals that make up the overarching goals of the power grid itself. These modules communicate with each other, exposing specific variables that other modules can set as needed. System performance is guaranteed by each module performing its assigned goals. The authors consider a case study where a wind generator module is brought into such a power grid. A modular model architecture would be especially applicable in open-access grids or microgrids, where many independent users both produce and consume electricity.

Giustolisi et al. [79] present a simulation method for analyzing the behavior of water distribution networks (WDNs), based on pressure as a metric of functionality. During operation, a WDN's topology may change as valves open and close or if a pipe bursts. This simulation can identify nodes in the network that are disconnected or fail to retain adequate pressure when the network topology changes. The technique can be used on very large networks, where exhaustively enumerating all topological changes is mathematically intractable.

Lin et al. [130] present an integrated cyber-physical simulator for WDNs, based on EPANET and MATLAB for simulation of the physical and cyber infrastructures, respectively. The physical portion of the network is simulated using EPANET, while the cyber portion is simulated using algorithms implemented in MATLAB. The simulator can facilitate identification of near-optimal controller settings. Lin et al. [132] apply game theory to provide decision support for water distribution.

Hasan et al. [90] develop a power grid simulator that incorporates control and protection element models. These models allow them to simulate faults in distance relays, overcurrent relays, and circuit breakers. The simulator is developed in Simulink, allowing timed sequences of both cyber and physical faults. Timing contingencies for $N - k$ contingency analysis is done manually but could be automated.

A refinement approach can also be applied to simulation, as in [31]. Initially the authors model both a controller and a physical plant using Sequential Function Charts, a nondeterminstic timed automata language. The plant model is constructed in a modular fashion, allowing re-use and simplifying the modeling process. These models can be co-simulated to verify the controller's behavior. Both models are then converted to PLC ladder logic, producing both the control software and a model of the plant that can be used to validate that software. These are executed on two PLCs, demonstrating the faithfulness of the controller model.

In *agent-based modeling*, the behavior of one or more intelligent components is abstracted as an autonomous agent. This abstraction, which facilitates system-level analysis, has been demonstrated to be useful in modeling complex distributed systems, and holds promise for overcoming the challenge of modeling heterogeneous CPS components. Recent studies demonstrate the use of this technique in modeling complex distributed systems.

Two seminal tutorials, [137, 138], provide a comprehensive introduction to agent-based modeling and simulation (ABMS). They describe the theoretical and practical foundations of ABMS, identify toolkits and methods for developing ABMS models, and compare ABMS with traditional modeling techniques.

A method for modeling dynamic and intelligent reconfiguration of real-time distributed control systems is introduced in [27]. These systems incorporate intelligent entities in the control network to enable quick reconfiguration—an ability that yields resilient CPSs. ABMS is particularly well-suited to modeling this type of system since each intelligent entity can be described as an autonomous agent. The model proposed by Nguyen et al. [159] represents both the logical connections among agents and their physical location to capture cyber and physical interdependence in both discrete- and continuous-event systems.

Lin et al. [128, 131, 133] propose a method for modeling interoperation among distributed, heterogeneous agents. They describe the semantics of agent data and use an ontology to define how data from different agents can be related. Data interoperability allows agents developed by several entities to be combined into models of large-scale CPSs. Their work also forms a foundation for modeling data flow, data corruption, and data cleansing in CPSs.

3.6 Formal Verification

Another take on modeling and analyzing control systems with very large or infinite state spaces is given in [41]. The authors propose a formal verification technique to model the system using linear temporal logic. Monte Carlo simulation is then used to determine the probability of system failure, even if the failure event is very rare.

Laibinis et al. [117] model the resilience of data processing systems of a CPS. Such systems need to process large amounts of data in parallel and withstand failure of nodes in the system. They model complex processing systems in Event-B [1], a modeling formalism that can capture both the initial configuration of the system as well as dynamic reconfigurations. Each system is checked using statistical model checking to determine how likely it is to successfully process data.

In *aspect-oriented programming*, each concern of the system, such as control, schedulability, scalability, and optimization, is considered an *aspect* and the system is represented in terms of these aspects. This enables the creation of simple models that encompass a single aspect of the system, resulting in a higher-level separation of concerns than traditional object-oriented programming allows. Aspects can subsequently be combined to create a complete representation of the system.

Zhang [224] proposes an aspect-oriented formal specification approach for real-time CPSs where various formal specification languages are semantically integrated instead of using a single specification technique for all concerns. The author also provides some verification techniques. Related verification techniques are proposed in [223] and [134], where the authors present an aspect-oriented model-driven architecture development method for non-functional requirements of CPSs. Aspects are identified as crosscutting non-functional properties of the system. The development process is driven by first generating a platform-independent model of the system, considering the functional requirements, and then weaving the generic aspects of the target system into the model to produce the platform-specific model.

Zhang and He [225] propose an aspect-oriented method for specifying quality of service requirements for CPSs. It uses a combination of UML, Real Time Logic (RTL), and Extended RTL (ERTL) which specify both relative and absolute timing in a real-time CPS. The resulting model uses both formal and semi-formal notations that provide a basis for a rigorous and practical QoS modeling.

Additionally, [76] present a formal, model-checking algorithm to evaluate the survivability of fluid critical infrastructures. Their procedure recursively traverses the state-space of the model, and identifies regions that satisfy a Stochastic Time Logic formula. This can be used to direct repair and maintenance as well as identify areas of the system requiring improvement.

A significant difficulty in CPS modeling is representation, in a unified model, of both continuous physical world and discrete cyber events. A gap inevitably arises between the discrete sensing and actuation ability of the control infrastructure and the continuous nature of physical events. This gap may cause the control infrastructure to fail to sense certain events or to violate the safety of the system by failing to control it properly. CPS modeling must capture both continuous physical and discrete cyber events in order to verify system safety. Susuki et al. [195] propose the use of hybrid system theory to address this challenge. In their approach, operational system states are identified and reachability analysis is used to ensure that only these operational states are reachable. Their analysis focuses on transient system stability, but the technique is applicable to larger systems and longer time durations.

Sun et al. [194] build a model of a smart grid in RT-PROMELA and use the model-checking software RT-SPIN [205] to verify that overall system properties hold. In order to ensure that the model-checking problem remains tractable, the system is decomposed into modules that are individually modeled and checked. The composition of these models is then checked to ensure interaction among modules does not cause violations of correctness or related system properties. The authors focus on physical sampling rates, ensuring that the sampling frequency of each module is sufficiently high to sense all events.

Unlike critical infrastructure CPS systems, medical devices are significantly more difficult to test during development. One approach to testing such systems is simultaneous simulation of both the medical device and the organ it is controlling. Jiang et al. [103] develop a virtual heart model for verifying pacemaker designs. The heart is modeled using both a geometric model and a temporal logic model. These models are overlaid onto a diagram of the heart, providing information about the propagation time of signals within the heart. Temporal logic is used to formally model the pacemaker, which allows the two models to be combined. The correctness and completeness of the proposed models are verified using closed-loop case studies.

Testing medical devices in real-world situations is difficult and highly regulated, and as such, model-driven design is a common approach to building medical CPSs. Detailed model-checking of all software used in such devices is necessary to ensure high confidence in their operation [123]. Interoperability with a broad range of emerging and legacy medical devices is critical, as is correctness of operation. Finally, a host of legal regulations govern and introduce significant constraints on the design and operation of these devices. Murugesan et al. [156] use qualitative architecture modeling and continuous-time quantitative modeling to capture design requirements and ensure that they are met. Li et al. [125] use formal verification to ensure that a pacemaker behaves as intended. They verify both the outputs of the system and the timing of those outputs and demonstrate the ability of their verification technique to find vulnerabilities in the system.

Formal verification results can be used to determine simulation parameters, ensuring simulation accuracy and assisting developers in writing specifications for physical systems [45]. The authors demonstrate this by constructing a real-time model of an HVAC controller and network protocol which is formally verified to meet certain timing and communication properties. The verification process determines what reasonable timing constraints are; these constraints are then used in a simulation of the whole system to determine the behavior of the physical HVAC plant under worst-case conditions.

Verification can be applied to dependability requirements as well. Drozdov et al. [52] study a fault location and isolation system for a power grid. This system is verified using a closed-loop model that incorporates both the controller and a model of the physical grid. The controller model alone has 2^{77} reachable states; adding the physical model reduces this to 2^{8} by eliminating impossible physical states that the controller model cannot know about. With this model it is possible to verify conditions such as “if a fault occurs, it will be isolated” and “if a circuit breaker trips, a corresponding switch will isolate the fault”. Furthermore, it can be verified that circuit breakers always trip immediately after a fault occurs.

A challenge to formal verification is that verification languages are unfamiliar to control system programmers. Sirjani et al. [191] develop Lingua Franca, a language similar to many common programming languages that can be compiled both to standard programming languages and to Timed Rebeca, a verification language. They construct a model of an automated train which contains several controllers and verify that it is safe. This model is able to demonstrate that ignoring the time actions take to complete in the physical world can lead to transient unsafe states where several actions occur in a non-deterministic order. Adding time delays to actions to synchronize them eliminates these states.

Much of CPS verification is done using model checking tools; however, verifying complex predicates on large systems can be intractable. Theorem proving tools can avoid these issues at the cost of being only partly automated. Traditionally the realm of certain mathematics and software engineering fields, they have more recently begun to be applied to CPSes. Rashid et al. [177] survey many of these applications and compare the expressiveness of various tools. While these tools are quite powerful, designers must develop theories of discrete- and continuous-time system behavior before they can apply these tools to verification. Of the theorem provers surveyed, only KeYmaera [172] already has theories for hybrid systems and is thus readily applicable to CPS verification.

4 OPEN QUESTIONS

5 CONCLUSION

Cyber-physical systems present many challenges to be addressed by modeling and design techniques. Among these challenges are interdependencies among components, the large-scale nature of CPSs, real-time processing and actuation constraints, and providing trustworthy service. As part of addressing these challenges, various non-functional attributes of CPSs are defined to capture the dependability of service in the face of component faults. In this paper, we describe numerous techniques for modeling both functional and non-functional attributes of CPSs. Furthermore, we investigate design approaches that address the challenges of building CPSs that are dependable and performant. In addition to modeling and design techniques that apply to any CPS, we provide case studies in the smart grid, water distribution network, medical, collaborative robotics, intelligent transportation, and intelligent and reconfigurable manufacturing domains. These case studies provide examples of how generic techniques can be applied to specific systems and address concerns specific to each domain. The works in this paper provide a foundation for building both dependable and performant CPSs.

REFERENCES

[1] Jean-Raymond Abrial. 2010. *Modeling in Event-B: system and software engineering.* Cambridge University Press, Cambridge, United Kingdom.

[2] Magan Lal Agrawal, Rajesh Gupta, and Pramod R. Bhave. 2007. Reliability-Based Strengthening and Expansion of Water Distribution Networks. *Journal of Water Resources Planning and Management* 133 (November/December 2007), 531–541.

[3] Amirreza Hassani Ahangar, Behrooz Vahidi, and Hossein Askarian Abyaneh. 2018. Evaluating smart grid reliability based on impacts of cyber (control, monitoring and protection) network and its different topologies. *International Journal of System Assurance Engineering and Management* 9, 5 (2018), 1047–1056.

[4] Ahmad Al-Hammouri, Vincenzo Liberatore, Huthaifa Al-Omari, Zakaria Al-Qudah, Michael S. Branicky, and Deepak Agrawal. 2007. A co-simulation platform for actuator networks. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems SenSys '07* (Sydney, Australia). ACM, New York, NY, USA, 383–384.

[5] Ahmad T Al-Hammouri. 2012. A comprehensive co-simulation platform for cyber-physical systems. *Computer Communications* 36, 1 (2012), 8–19.

[6] Murtadha N Albasrawi, Nathan Jarus, Kamlesh A Joshi, and Sahra Sedigh Sarvestani. 2014. Analysis of reliability and resilience for smart grids. In *Computer Software and Applications Conference (COMPSAC), 2014 IEEE 38th Annual* (Vasteras, Sweden). IEEE, Washington, DC, USA, 529–534.

[7] Gisliany Alves, Danielle Marques, Ivanovitch Silva, Luiz Affonso Guedes, and Maria da Guia da Silva. 2019. A Methodology for Dependability Evaluation of Smart Grids. *Energies* 12, 9 (May 2019), 1817. https://doi.org/10.3390/en12091817

[8] F. Aminifar, M. Fotuhi-Firuzabad, and R. Billinton. 2007. Extended reliability model of a unified power flow controller. *Generation, Transmission, and Distribution, IET* 1, 6 (November 2007), 896–903.

[9] Eva Andrijcic and Yacov Y. Haimes. 2017. Metamodeling of Interdependent Systems: Application to Bridge Infrastructure Management. *Journal of Infrastructure Systems* 23, 2 (June 2017), 04016028. https://doi.org/10.1061/(ASCE)IS.1943-555X.0000322

[10] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. 2004. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing* 1, 1 (2004), 11–33. https://doi.org/10.1109/TDSC.2004.2

[11] Alberto Avritzer, Felicita Di Giandomenico, Anne Remke, and Martin Riedl. 2012. Assessing Dependability and Resilience in Critical Infrastructures: Challenges and Opportunities. In *Resilience Assessment and Evaluation of Computing Systems*, Katinka Wolter, Alberto Avritzer, Marco Vieira, and Aad van Moorsel (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 41–63. https://doi.org/10.1007/978-3-642-29032-9_3

[12] Adnan Aziz, Kumud Sanwal, Vigyan Singhal, and Robert Brayton. 2000. Model-checking Continuous-time Markov Chains. *ACM Trans. Comput. Logic* 1, 1 (July 2000), 162–170. https://doi.org/10.1145/343369.343402

[13] Joydeep Banerjee, Kaustav Basu, and Arunabha Sen. 2018. On hardening problems in critical infrastructure systems. *International Journal of Critical Infrastructure Protection* 23 (Dec. 2018), 49–67. https://doi.org/10.1016/j.ijcip.2018.08.001

[14] Kash Barker, Jose Emmanuel Ramirez-Marquez, and Claudio M. Rocco. 2013. Resilience-based network component importance measures. *Reliability Engineering & System Safety* 117 (Sept. 2013), 89–97. https://doi.org/10.1016/j.ress.2013.03.012

[15] Richard E Barlow, Jerry B Fussell, and Nozer D Singpurwalla. 1975. *Reliability and fault tree analysis.* Vol. 33. Society for Industrial and Applied Mathematics, Philadelphia, PA.

[16] Davide Basile, Felicita Di Giandomenico, and Stefania Gnesi. 2018. A Refinement Approach to Analyse Critical Cyber-Physical Systems. In *Software Engineering and Formal Methods (Lecture Notes in Computer Science)*, Antonio Cerone and Marco Roveri (Eds.). Springer International Publishing, Cham, 267–283.

[17] Marco Beccuti, Silvano Chiaradonna, Felicita Di Giandomenico, Susanna Donatelli, Giovanna Dondossola, and Giuliana Franceschinis. 2012. Quantification of dependencies between electrical and information infrastructures. *International Journal of Critical Infrastructure Protection* 5, 1 (2012), 14 – 27. https://doi.org/10.1016/j.ijcip.2012.01.003

[18] Kristian Beckers, Isabelle Côté, Thomas Frese, Denis Hatebur, and Maritta Heisel. 2015. A Structured Validation and Verification Method for Automotive Systems Considering the OEM/Supplier Interface. In *Computer Safety, Reliability, and Security - 34th International Conference, SAFECOMP 2015 Delft, The Netherlands, September 23-25, 2015. Proceedings.* Springer International, Cham, 90–108.

[19] R. G. Bennetts. 1982. Analysis of Reliability Block Diagrams by Boolean Techniques. *IEEE Transactions on Reliability* R-31, 2 (June 1982), 159–166. https://doi.org/10.1109/TR.1982.5221283

[20] Betty Biringer, Eric Vugrin, Drake Warren, Eric Vugrin, and Drake Warren. 2013. *Critical Infrastructure System Security and Resiliency.* CRC Press, Boca Raton. https://doi.org/10.1201/b14566

[21] Simon Bliudze, Sébastien Furic, Joseph Sifakis, and Antoine Viel. 2019. Rigorous design of cyber-physical systems. *Software & Systems Modeling* 18, 3 (June 2019), 1613–1636. https://doi.org/10.1007/s10270-017-0642-5

[22] H. Bohnenkamp, T. Courtney, D. Daly, S. Derisavi, H. Hermanns, J. Katoen, R. Klaren, Vinh Vi Lam, and W.H. Sanders. 2003. On integrating the Möbius and MODEST modeling tools. In *Dependable Systems and Networks, 2003. Proceedings. 2003 International Conference on.* IEEE Computer Society Press, Washington, DC, USA, 671–671. https://doi.org/10.1109/DSN.2003.1209980

[23] Victor Bolbot, Gerasimos Theotokatos, Luminita Manuela Bujorianu, Evangelos Boulougouris, and Dracos Vassalos. 2019. Vulnerabilities and safety assurance methods in Cyber-Physical Systems: A comprehensive review. *Reliability Engineering & System Safety* 182 (Feb. 2019), 179–193. https://doi.org/10.1016/j.ress.2018.09.004

[24] Borzoo Bonakdarpour. 2008. Challenges in transformation of existing real-time embedded systems to cyber-physical systems. *ACM SIGBED Review* 5, 1 (2008), 1–2. https://doi.org/10.1145/1366283.1366294

[25] Michael V. Boutsikas and Markos V. Koutras. 2000. Reliability approximation for Markov chain imbeddable systems. *Methodology and Computing in Applied Probability* 2, 4 (2000), 393–411.

[26] Justin M. Bradley and Ella M. Atkins. 2012. Toward Continuous State-Space Regulation of Coupled Cyber-Physical Systems. *Proc. IEEE* 100, 5985457 (January 2012), 60–74. Issue 1.

[27] Robert W. Brennan, Martyn Fletcher, and Douglas H. Norrie. 2002. An Agent-Based Approach to Reconfiguration of Real-Time Distributed Control Systems. *IEEE Transactions on Robotics and Automation* 18 (August 2002), 441–451.

[28] Christopher Brooks, Thomas H. Feng, Edward A. Lee, and Reinhard van Hanxleden. 2008. *Multimodeling: A Preliminary Case Study.* Technical Report. Defense Technical Information Center.

[29] M.C. Bujorianu and H. Barringer. 2009. An Integrated Specification Logic for Cyber-Physical Systems. In *Proceedings of the 14th IEEE International Conference on Engineering of Complex Computer Systems.* IEEE, Los Alamitos, CA, USA, 291 –300. https://doi.org/10.1109/ICECCS.2009.36

[30] Gabrio Caimi, Martin Fuchsberger, Marco Laumanns, and Marco Lüthi. 2012. A model predictive control approach for discrete-time rescheduling in complex central railway station areas. *Computers & Operations Research* 39, 11 (2012), 2578–2593.

[31] Nuno Canadas, José Machado, Filomena Soares, Carlos Barros, and Leonilde Varela. 2018. Simulation of cyber physical systems behaviour using timed plant models. *Mechatronics* 54 (Oct. 2018), 175–185. https://doi.org/10.1016/j.mechatronics.2017.10.009

[32] Janette Cardoso, Patricia Derler, John C. Eidson, and Edward A. Lee. 2011. Network Latency and Packet Delay Variation in Cyber-physical Systems. In *Proceedings of the Network Science Workshop(NSW'11), IEEE* (West Point, NY). IEEE, Washington, DC, USA, 51–58.

[33] B. A. Carreras, D. E. Newman, I. Dobson, and J. M. Reynolds Barredo. 2016. The Impact of Local Power Balance and Link Reliability on Blackout Risk in Heterogeneous Power Transmission Grids. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, Kauai, HI, USA, 2584–2593. https://doi.org/10.1109/HICSS.2016.322 ISSN: 1530-1605.

[34] Jean-Francois Castet and Joseph H Saleh. 2012. On the concept of survivability, with application to spacecraft and space-based networks. *Reliability Engineering & System Safety* 99 (2012), 123–138.

[35] Stathis Chadjiconstantinidis and Markos V. Koutras. 1999. Measures of component importance for Markov chain imbeddable reliability structures. *Naval Research Logistics (NRL)* 46, 6 (1999), 613–639.

[36] Ramesh Chandra, R.M. Lefever, Michel Cukier, and W.H. Sanders. 2000. Loki: a state-driven fault injector for distributed systems. In *Dependable Systems and Networks, 2000. DSN 2000. Proceedings International Conference on*. IEEE, Washington, DC, USA, 237–242. https://doi.org/10.1109/ICDSN.2000.857544

[37] Octav Chipara and Chenyang Lu. 2008. Towards predictable wireless cyber-physical applications. *ACM SIGBED Review* 5, 1 (2008), 1–2. https://doi.org/10.1145/1366283.1366298

[38] Xiaodong Chu, Rongxiang Zhang, Maosen Tang, Haoyi Huang, and Lei Zhang. 2018. Cyber Physical System Modelling of Distribution Power Systems for Dynamic Demand Response. *IOP Conference Series: Earth and Environmental Science* 111 (Jan. 2018), 012029. https://doi.org/10.1088/1755-1315/111/1/012029

[39] Gianfranco Ciardo, Raymond A. Marie, Bruno Sericola, and Kishor S. Trivedi. 1990. Performability analysis using semi-Markov reward processes. *Computers, IEEE Transactions on* 39, 10 (1990), 1251–1264.

[40] G. Clark, T. Courtney, D. Daly, D. Deavours, S. Derisavi, J.M. Doyle, W.H. Sanders, and P. Webster. 2001. The Möbius modeling tool. In *Petri Nets and Performance Models, 2001. Proceedings. 9th International Workshop on*. IEEE, Washington, DC, USA, 241–250. https://doi.org/10.1109/PNPM.2001.953373

[41] Edmund M. Clarke and Paolo Zuliani. 2011. Statistical Model Checking for Cyber-Physical Systems. In *Proceedings of the 9th International Symposium on Automated Technology for Verification and Analysis(ATVA'11)* (Taipei). Springer Berlin Heidelberg, Berlin, Heidelberg, 1–12.

[42] Lucia Cloth and Boudewijn R Haverkort. 2005. Model checking for survivability!. In *Quantitative Evaluation of Systems, 2005. Second International Conference on the*. IEEE, Washington, DC, USA, 145–154.

[43] T. Courtney, D. Daly, S. Derisavi, S. Gaonkar, M. Griffith, V. Lam, and W.H. Sanders. 2004. The Möbius modeling environment: recent developments. In *Quantitative Evaluation of Systems, 2004. QEST 2004. Proceedings. First International Conference on the*. IEEE, Washington, DC, USA, 328–329. https://doi.org/10.1109/QEST.2004.1348051

[44] T. Courtney, S. Derisavi, S. Gaonkar, M. Griffith, V. Lam, M. McQuinn, E. Rozier, and W.H. Sanders. 2005. The Möbius Modeling Environment: Recent Extensions - 2005. In *Quantitative Evaluation of Systems, 2005. Second International Conference on the*. IEEE, Washington, DC, USA, 259–260. https://doi.org/10.1109/QEST.2005.39

[45] Luís Diogo Couto, Stylianos Basagiannis, El Hassan Ridouane, Alie El-Din Mady, Miran Hasanagic, and Peter Gorm Larsen. 2018. Injecting Formal Verification in FMI-Based Co-simulations of Cyber-Physical Systems. In *Software Engineering and Formal Methods (Lecture Notes in Computer Science)*, Antonio Cerone and Marco Roveri (Eds.). Springer International Publishing, Cham, 284–299.

[46] Fabio Cremona, Marten Lohstroh, David Broman, Edward A. Lee, Michael Masin, and Stavros Tripakis. 2019. Hybrid co-simulation: it's about time. *Software & Systems Modeling* 18, 3 (June 2019), 1655–1679. https://doi.org/10.1007/s10270-017-0633-6

[47] Tina Dasic and Branislav Djordjevic. 2004. Method for Water Distribution Systems Reliability Evaluation, In Proceedings of the 2nd International Congress on Environmental Modelling and Software. *International Environmental Modelling and Software Society* 1, 153–158.

[48] Maíra A. de C. Gatti, Carlos J. P. de Lucena, and Jean-Pierre Briot. 2007. On Fault Tolerance in Law-Governed Multi-Agent Systems. In *Software Engineering for Multi-Agent Systems V: Research Issues and Practical Applications*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1–20. https://doi.org/10.1007/978-3-540-73131-3_1

[49] P. Derler, E.A. Lee, and A.-S. Vincentelli. 2012. Modeling Cyber-Physical Systems. *Proc. IEEE* 100, 1 (2012), 13–28. https://doi.org/10.1109/JPROC.2011.2160929

[50] Henrique Dezani, Luis Gomes, Furio Damiani, and Norian Marranghello. 2012. Controlling traffic jams on urban roads modeled in Coloured Petri net using Genetic Algorithm. In *IECON 2012-38th Annual Conference on IEEE Industrial Electronics Society*. IEEE, Washington, DC, USA, 3043–3048.

[51] DIgSILENT. 2017. DIgSILENT PowerFactory. http://www.digsilent.de/index.php/products-powerfactory.html. Retrieved Feb. 21, 2017.

[52] Dmitrii Drozdov, Sandeep Patil, Chen-Wei Yang, Gulnara Zhabelova, and Valeriy Vyatkin. 2018. Formal Verification of Protection Functions for Power Distribution Networks. In *44th Annual Conference of the IEEE Industrial Electronics*

Society (IECON 2018). IEEE, Washington, DC, 3550–3555. https://doi.org/10.1109/IECON.2018.8592802 ISSN: 2577-1647.

[53] L. Dueas-Osorio, J. I. Craig, and B. J. Goodno. 2004. Probabilistic response of interdependent infrastructure networks. In *Proceedings of the 2nd annual meeting of the Asian-pacific network of centers for earthquake engineering research ANCER '04* (Honolulu, Hawaii). Multidisciplinary Center for Earthquake Engineering Research, New York, NY, USA, 28–30.

[54] Danny Dyer. 1989. Unification of reliability/availability/repairability models for Markov systems. *Reliability, IEEE Transactions on* 38, 2 (1989), 246–252.

[55] Pedro R D'Argenio, Holger Hermanns, Joost-Pieter Katoen, and Ric Klaren. 2001. Modest–a modelling and description language for stochastic timed systems. In *Process Algebra and Probabilistic Methods. Performance Modelling and Verification*. Springer, Berlin, Germany, 87–104.

[56] John C. Eidson, Edward A. Lee, Slobodan Matic, Sanjit A. Seshia, and Jia Zou. 2012. Distributed Real-Time Software for Cyber-Physical Systems. *Proc. IEEE* 100, 5995282 (January 2012), 45–59. Issue 1.

[57] Robert J Ellison, David A Fisher, Richard C Linger, Howard F Lipson, and Thomas Longstaff. 1997. *Survivable network systems: An emerging discipline*. Technical Report. DTIC Document.

[58] Electric Power Research Institute (EPRI). 2008. OpenDSS. https://www.epri.com/pages/sa/opendss. Retrieved August 7, 2020.

[59] Irene Eusgeld, Cen Nan, and Sven Dietz. 2011. "System-of-systems" approach for interdependent critical infrastructures. *Reliability Engineering & System Safety* 96, 6 (June 2011), 679–686. https://doi.org/10.1016/j.ress.2010.12.010

[60] Barry C. Ezell, John V. Farr, and Ian Wiese. 2000. Infrastructure Risk Analysis Model. *Journal of Infrastructure Systems* 6 (September 2000), 114–117.

[61] Zhong Fan, Parag Kulkarni, Sedat Gormus, Costas Efthymiou, Georgios Kalogridis, Mahesh Sooriyabandara, Ziming Zhu, Sangarapillai Lambotharan, and Woon Hau Chin. 2013. Smart grid communications: Overview of research challenges, solutions, and standardization activities. *Communications Surveys & Tutorials, IEEE* 15, 1 (2013), 21–38.

[62] M. P. Fanti, G. Iacobellis, A. M. Mangini, and W. Ukovich. 2014. Freeway Traffic Modeling and Control in a First-Order Hybrid Petri Net Framework. *IEEE Transactions on Automation Science and Engineering* 11, 1 (Jan 2014), 90–102. https://doi.org/10.1109/TASE.2013.2253606

[63] Ayman Faza, Sahra Sedigh, and Bruce McMillin. 2010. *Integrated cyber-physical fault injection for reliability analysis of the smart grid*. Springer Berlin Heidelberg, Berlin, Heidelberg. 277–290 pages.

[64] Ayman Z. Faza, Sahra Sedigh, and Bruce M. Mcmillin. 2007. Reliability modeling for the advanced electric power grid. In *Proceedings of the 26th International Conference on Computer Safety, Reliability, and Security SAFECOMP '07* (Nuremberg, Germany). Springer Berlin Heidelberg, Berlin, Heidelberg, 370–383. https://doi.org/10.1007/978-3-540-75101-4_35

[65] Ayman Z. Faza, Sahra Sedigh, and Bruce M. Mcmillin. 2008. The Advanced Electric Power Grid: Complexity Reduction Techniques for Reliability Modeling. In *Proceedings of the 27th international conference on Computer Safety, Reliability, and Security SAFECOMP '08*. Springer Berlin Heidelberg, Berlin, Heidelberg, 429–439.

[66] Ayman Z. Faza, Sahra Sedigh, and Bruce M. Mcmillin. 2009. Reliability Analysis for the Advanced Electric Power Grid: From Cyber Control and Communication to Physical Manifestations of Failure. In *Proceedings of the 28th International Conference on Computer Safety, Reliability, and Security SAFECOMP '09*. Springer Berlin Heidelberg, Berlin, Heidelberg, 257–269.

[67] Peter H. Feiler, David P. Gluch, and John J. Hudak. 2006. *The architecture analysis & design language (AADL): An introduction*. Technical Report. DTIC Document.

[68] Roberto Filippini and Andrés Silva. 2014. A modeling framework for the resilience analysis of networked systems-of-systems based on functional dependencies. *Reliability Engineering & System Safety* 125 (2014), 82–91.

[69] Dale Fitch, Sahra Sedigh, Bruce McMillin, and Ravi Akella. 2013. CPS-CSH cyber-physical analysis and design. In *Critical Information Infrastructures Security*. Springer Berlin Heidelberg, Berlin, Heidelberg, 92–105.

[70] John Fitzgerald, Carl Gamble, Richard Payne, Peter Gorm Larsen, Stylianos Basagiannis, and Alie El-Din Mady. 2016. Collaborative Model-based Systems Engineering for Cyber-Physical Systems, with a Building Automation Case Study. In *INCOSE International Symposium*, Vol. 26. Wiley Online Library, Hoboken, NJ, USA, 817–832. http://onlinelibrary.wiley.com/doi/10.1002/j.2334-5837.2016.00195.x/full

[71] J. Fitzgerald, K. Pierce, and C. Gamble. 2012. A rigorous approach to the design of resilient cyber-physical systems through co-simulation. In *Dependable Systems and Networks Workshops (DSN-W), 2012 IEEE/IFIP 42nd International Conference on*. IEEE, Washington, DC, USA, 1–6.

[72] Giulio Galvan and Jitendra Agarwal. 2017. Community Detection in Action: Identification of Critical Elements in Infrastructure Networks. *Journal of Infrastructure Systems* 24, 1 (2017), 04017046.

[73] Giulio Galvan and Jitendra Agarwal. 2020. Assessing the vulnerability of infrastructure networks based on distribution measures. *Reliability Engineering & System Safety* 196 (April 2020), 106743. https://doi.org/10.1016/j.ress.2019.

106743

[74] Shravan Gaonkar, Ken Keefe, Ruth Lamprecht, Eric Rozier, Peter Kemper, and William H. Sanders. 2009. Performance and Dependability Modeling with Möbius. *SIGMETRICS Perform. Eval. Rev.* 36, 4 (March 2009), 16–21. https://doi.org/10.1145/1530873.1530878

[75] Hamed Ghasemieh, Anne Remke, and Boudewijn R. Haverkort. 2013. Analysis of a Sewage Treatment Facility Using Hybrid Petri Nets. In *Proceedings of the 7th International Conference on Performance Evaluation Methodologies and Tools* (Torino, Italy) *(ValueTools '13)*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), Brussels, BEL, 165–174. https://doi.org/10.4108/icst.valuetools.2013.254384

[76] Hamed Ghasemieh, Anne Remke, and Boudewijn R Haverkort. 2013. Survivability evaluation of fluid critical infrastructures using hybrid Petri nets. In *Dependable Computing (PRDC), 2013 IEEE 19th Pacific Rim International Symposium on*. IEEE, IEEE, Vancouver, BC, Canada, 152–161.

[77] A. Gheisi, M. Forsyth, and Gh Naser. 2016. Water Distribution Systems Reliability: A Review of Research Literature. *Journal of Water Resources Planning and Management* 142, 11 (2016), 04016047.

[78] Rahul Ghosh, DongSeong Kim, and Kishor S Trivedi. 2013. System resiliency quantification using non-state-space and state-space analytic models. *Reliability Engineering & System Safety* 116 (2013), 109–125.

[79] Orazio Giustolisi, Zoran Kapelan, and Dragan Savic. 2008. Algorithm for Automatic Detection of Topological Changes in Water Distribution Networks. *Journal of Hydraulic Engineering* 134 (2008), 435–446.

[80] Antoon Goderis, Christopher Brooks, Ilkay Altintas, Edward A. Lee, and Carole Goble. 2009. Heterogeneous composition of models of computation. *Future Generation Computer Systems* 25, 5 (May 2009), 552–560. https://doi.org/10.1016/j.future.2008.06.014

[81] Ambuj Goyal, Stephen S Lavenberg, and Kishor Shridharbhai Trivedi. 1987. Probabilistic modeling of computer system availability. *Annals of Operations Research* 8, 1 (1987), 285–306.

[82] D Grether, Andreas Neumann, and Kai Nagel. 2012. Simulation of urban traffic control: A queue model approach. *Procedia Computer Science* 10 (2012), 808–814.

[83] T1A1.2 Working group. 2001. *Technical Report on Enhanced Network Survivability Performance*. Technical Report 68. Alliance for Telecommunications Industry Solutions (ATIS).

[84] Volkan Gunes, Steffen Peter, Tony Givargis, and Frank Vahid. 2014. A Survey on Concepts, Applications, and Challenges in Cyber-Physical Systems | Cyber Physical Systems Design Group. *KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS* 8, 12 (2014), 4242–4268. http://cps.ics.uci.edu/publication/cps_survey/

[85] Yacov Y. Haimes. 2005. Infrastructure Interdependencies and Homeland Security. *Journal of Infrastructure Systems* 11 (June 2005), 65–66.

[86] Yacov Y. Haimes and Barry M. Horowitz. 2004. Modeling Interdependent Infrastructures for Sustainable Counterterrorism. *Journal of Infrastructure Systems* 10 (June 2004), 33–42.

[87] Yacov Y. Haimes, Nicholas C. Matalas, and James H. Lambert. 1998. Reducing Vulnerability of Water Supply Systems to Attack. *Journal of Infrastructure Systems* 4 (December 1998), 164–177.

[88] Wang Hanbo, Zhou Xingshe, Dong Yunwei, and Tang Lei. 2009. Modeling Timing Behavior for Cyber-Physical Systems. In *Proceedings of International Conference on Computational Intelligence and Software Engineering CiSE '09*. IEEE, Washington, DC, USA, 1–4.

[89] Ali-Mohammad Hariri, Hamed Hashemi-Dezaki, and Maryam A. Hejazi. 2020. A novel generalized analytical reliability assessment method of smart grids including renewable and non-renewable distributed generations and plug-in hybrid electric vehicles. *Reliability Engineering & System Safety* 196 (April 2020), 106746. https://doi.org/10.1016/j.ress.2019.106746

[90] Saqib Hasan, Ajay Chhokra, Abhishek Dubey, Nagabhushan Mahadevan, Gabor Karsai, Rishabh Jain, and Srdjan Lukic. 2017. A simulation testbed for cascade analysis, In 2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT). *IEEE PES Innovative Smart Grid Technologies (ISGT)*, 1–5. http://www.isis.vanderbilt.edu/sites/default/files/ISGT_2017_paper1.pdf

[91] M. Hecht, A. Lam, and C. Vogl. 2011. A Tool Set for Integrated Software and Hardware Dependability Analysis Using the Architecture Analysis and Design Language (AADL) and Error Model Annex. In *2011 16th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS)*. IEEE, Washington, DC, USA, 361–366. https://doi.org/10.1109/ICECCS.2011.44

[92] Poul E Heegaard and Kishor S Trivedi. 2009. Network survivability modeling. *Computer Networks* 53, 8 (2009), 1215–1234.

[93] Devanandham Henry and Jose Emmanuel Ramirez-Marquez. 2012. Generic metrics and quantitative approaches for system resilience as a function of time. *Reliability Engineering & System Safety* 99 (2012), 114–122.

[94] Mohammad Mehdi Hosseini and Masood Parvania. 2020. Quantifying impacts of automation on resilience of distribution systems. *IET Smart Grid* 3, 2 (Feb. 2020), 144–152. https://doi.org/10.1049/iet-stg.2019.0175 Publisher: IET Digital Library.

[95] Liang Hu, Nannan Xie, Zhejun Kuang, and Kuo Zhao. 2012. Review of Cyber-Physical System Architecture. In *2012 15th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW)*. IEEE, Washington, DC, USA, 25–30. https://doi.org/10.1109/ISORCW.2012.15

[96] Zhen Huang, Cheng Wang, Milos Stojmenovic, and Amiya Nayak. 2013. Balancing system survivability and cost of smart grid via modeling cascading failures. *IEEE Transactions on Emerging Topics in Computing* 1, 1 (2013), 45–56. https://doi.org/10.1109/TETC.2013.2273079

[97] Zhen Huang, Cheng Wang, Milos Stojmenovic, and Amiya Nayak. 2015. Characterization of Cascading Failures in Interdependent Cyber-Physical Systems. *IEEE Trans. Comput.* 64, 8 (August 2015), 2158–2168.

[98] Abdulmalik Humayed, Jingqiang Lin, Fengjun Li, and Bo Luo. 2017. Cyber-Physical Systems Security—A Survey. *IEEE Internet of Things Journal* 4, 6 (Dec. 2017), 1802–1831. https://doi.org/10.1109/JIOT.2017.2703172 Conference Name: IEEE Internet of Things Journal.

[99] Marija D. Ilic, Le Xie, Usman A. Khan, and José M.F. Moura. 2008. Modeling future cyber-physical energy systems. In *Proceedings of the IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*. IEEE, Washington, DC, USA, 1–9. https://doi.org/10.1109/PES.2008.4596708

[100] B.R. Iyer, L. Donatiello, and P. Heidelberger. 1986. Analysis of Performability for Stochastic Models of Fault-Tolerant Systems. *Computers, IEEE Transactions on* C-35, 10 (Oct 1986), 902–907.

[101] Nicholas Jacobs, Shamina Hossain-McKenzie, and Eric Vugrin. 2018. Measurement and Analysis of Cyber Resilience for Control Systems: An Illustrative Example. In *2018 Resilience Week (RWS)*. IEEE, Denver, CO, USA, 38–46. https://doi.org/10.1109/RWEEK.2018.8473549

[102] Hyung Seok Jeong and Dulcy M. Abraham. 2006. Operational Response Model for Physically Attacked Water Networks Using NSGA-II. *Journal of Computing in Civil Engineering* 20 (September/October 2006), 328–338.

[103] Zhihao Jiang, Miroslav Pajic, and Rahul Mangharam. 2012. Cyber-Physical Modeling of Implantable Cardiac Medical Devices. *Proc. IEEE* 100 (January 2012), 122–137. Issue 1.

[104] Jorge Julvez and Rene K Boel. 2010. A continuous Petri net approach for model predictive control of traffic systems. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 40, 4 (2010), 686–697.

[105] M. Kaegi, R. Mock, and W. Kroger. 2009. Analyzing maintenance strategies by agent based simulations: A feasibility study. *Reliability Engineering and Systems Safety* 94, 9 (2009), 1416 – 1421. https://doi.org/10.1016/j.ress.2009.02.002

[106] Igor Kaitovic, Slobodan Lukovic, and Miroslaw Malek. 2015. Unifying Dependability of Critical Infrastructures: Electric Power System and ICT: Concepts, Figures of Merit and Taxonomy. In *2015 IEEE 21st Pacific Rim International Symposium on Dependable Computing (PRDC)*. IEEE, Zhangjiajie, China, 50–59. https://doi.org/10.1109/PRDC.2015.38

[107] Polinpapilinho F. Katina, C. Ariel Pinto, Joseph M. Bradley, and Patrick T. Hester. 2014. Interdependency-induced risk with applications to healthcare. *International Journal of Critical Infrastructure Protection* 7, 1 (March 2014), 12–26. https://doi.org/10.1016/j.ijcip.2014.01.005

[108] Y.A. Katsigiannis, P.S. Georgilakis, and G.J. Tsinarakis. 2010. A Novel Colored Fluid Stochastic Petri Net Simulation Model for Reliability Evaluation of Wind/PV/Diesel Small Isolated Power Systems. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 40, 6 (Nov 2010), 1296–1309.

[109] J Kelly, T Ersal, C Li, B Marshall, S Kundu, G Keoleian, H Peng, I Hiskens, and J Stein. 2015. Sustainability, Resiliency, and Grid Stability of the Coupled Electricity and Transportation Infrastructures: Case for an Integrated Analysis. *Journal of Infrastructure Systems* 0, 0 (2015), 04015001. https://doi.org/10.1061/(ASCE)IS.1943-555X.0000251

[110] Dong Seong Kim, Khaja Mohammad Shazzad, and Jong Sou Park. 2006. A framework of survivability model for wireless sensor network. In *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*. IEEE, IEEE, Washington, DC, USA, 8–pp.

[111] R Kissel. 2013. Glossary of Key Information Security Terms (NISTIR 7298).

[112] John C Knight, Elisabeth A Strunk, and Kevin J Sullivan. 2003. Towards a rigorous definition of information system survivability. In *DARPA Information Survivability Conference and Exposition, 2003. Proceedings*, Vol. 1. IEEE, IEEE, Washington, DC, USA, 78–89.

[113] Yakup Koç, Martijn Warnier, Robert E Kooij, and Frances MT Brazier. 2013. A robustness metric for cascading failures by targeted attacks in power networks. In *Networking, Sensing and Control (ICNSC), 2013 10th IEEE International Conference on*. IEEE, IEEE, Washington, DC, USA, 48–53.

[114] Philip Koopman and Michael Wagner. 2014. Transportation CPS safety challenges. In *NSF Workshop on Transportation Cyber Physical Systems*. Carnegie Mellon University. https://doi.org/10.1184/R1/6469496.v1

[115] Yakup Koç, Martijn Warnier, Robert E. Kooij, and Frances M.T. Brazier. 2013. An entropy-based metric to quantify the robustness of power grids against cascading failures. *Safety Science* 59 (Nov. 2013), 126–134. https://doi.org/10.1016/j.ssci.2013.05.006

[116] Alexis Kwasinski. 2016. Quantitative Model and Metrics of Electrical Grids' Resilience Evaluated at a Power Distribution Level. *Energies* 9, 2 (Feb. 2016), 93. https://doi.org/10.3390/en9020093

[117] Linas Laibinis, Dmitry Klionskiy, Elena Troubitsyna, Anatoly Dorokhov, Johan Lilius, and Mikhail Kupriyanov. 2014. Modelling Resilience of Data Processing Capabilities of CPS. In *Software Engineering for Resilient Systems*, István Majzik and Marco Vieira (Eds.). Number 8785 in Lecture Notes in Computer Science. Springer International Publishing, Cham, 55–70. http://link.springer.com/chapter/10.1007/978-3-319-12241-0_5

[118] Jean-Claude Laprie, Karama Kanoun, and Mohamed Kaâniche. 2007. Modelling interdependencies between the electricity and information infrastructures. In *Computer Safety, Reliability, and Security: 26th International Conference, SAFECOMP 2007, Nuremberg, Germany, September 18-21, 2007. Proceedings*. Vol. 4680. Springer Berlin Heidelberg, Berlin, Heidelberg, 54–67. https://doi.org/10.1007/978-3-540-75101-4_5

[119] Edward A. Lee. 2008. Cyber Physical Systems: Design Challenges. In *Proceedings of the 2008 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing, ISORC '08*. IEEE Computer Society, Washington, DC, USA, 363–369. https://doi.org/10.1109/ISORC.2008.25

[120] Edward A. Lee. 2009. Introducing embedded systems: a cyber-physical approach: extended abstract. In *Proceedings of the 2009 Workshop on Embedded Systems Education, WESS '09* (Grenoble, France). ACM, New York, NY, USA, 1–2. https://doi.org/10.1145/1719010.1719011

[121] Edward A. Lee. 2016. Fundamental Limits of Cyber-Physical Systems Modeling. *ACM Trans. Cyber-Phys. Syst.* 1, 1 (Nov. 2016), 3:1–3:26. https://doi.org/10.1145/2912149

[122] Edward A. Lee and Marjan Sirjani. 2018. What Good are Models?. In *Formal Aspects of Component Software (Lecture Notes in Computer Science)*, Kyungmin Bae and Peter Csaba Ölveczky (Eds.). Springer International Publishing, Cham, 3–31.

[123] Insup Lee, Oleg Sokolsky, Sanjian Chen, John Hatcliff, Eunkyoung Jee, BaekGyu Kim, Andrew King, Margaret Mullen-Fortino, Soojin Park, Alexander Roederer, and Krishna K. Venkatasubramanian. 2012. Challenges and Research Directions in Medical Cyber-Physical Systems. *Proc. IEEE* 100 (January 2012), 75–90. Issue 1.

[124] Wen-Shing Lee, Doris L Grosh, Frank A Tillman, and Chang H Lie. 1985. Fault Tree Analysis, Methods, and Applications — A Review. *Reliability, IEEE Transactions on* 34, 3 (1985), 194–203.

[125] C. Li, A. Raghunathan, and N. K. Jha. 2013. Improving the Trustworthiness of Medical Device Software with Formal Verification Methods. *IEEE Embedded Systems Letters* 5, 3 (Sept. 2013), 50–53. https://doi.org/10.1109/LES.2013.2276434

[126] Jing Lin, Ann Miller, and Sahra Sedigh. 2011. *Integrated cyber-physical simulation of intelligent water distribution networks.* INTECH Open Access Publisher, Rijeka, Croatia. http://cdn.intechopen.com/pdfs/17563/intech-integrated_cyber_physical_simulation_of_intelligent_water_distribution_networks.pdf

[127] Jing Lin, S. Sedigh, and A.R. Hurson. 2012. Ontologies and Decision Support for Failure Mitigation in Intelligent Water Distribution Networks. In *System Science (HICSS), 2012 45th Hawaii International Conference on*. IEEE, Washington, DC, USA, 1187–1196. https://doi.org/10.1109/HICSS.2012.458

[128] J. Lin, S. Sedigh, and A. R. Hurson. 2011. An Agent-Based Approach to Reconciling Data Heterogeneity in Cyber-Physical Systems. In *2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and PhD Forum (IPDPSW)*. IEEE, Washington, DC, USA, 93–103. https://doi.org/10.1109/IPDPS.2011.130

[129] Jing Lin, Sahra Sedigh, and Ann Miller. 2009. A General Framework for Quantitative Modeling of Dependability in Cyber-Physical Systems: A Proposal for Doctoral Research. In *Proceedings of the 33rd Annual IEEE International Computer Software and Applications Conference, COMPSAC '09*, Vol. 1. IEEE, Washington, DC, USA, 668 –671. https://doi.org/10.1109/COMPSAC.2009.103

[130] J. Lin, S. Sedigh, and A. Miller. 2009. Towards Integrated Simulation of Cyber-Physical Systems: A Case Study on Intelligent Water Distribution. In *Proceedings of the Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing DASC '09*. IEEE, Washington, DC, USA, 690 –695. https://doi.org/10.1109/DASC.2009.140

[131] J. Lin, S. Sedigh, and A. Miller. 2010. Modeling Cyber-Physical Systems with Semantic Agents. In *Proceedings of the 34th IEEE International Computer Software and Applications Conference COMPSAC '10* (Seoul, South Korea). IEEE, Washington, DC, USA, 13–18.

[132] J. Lin, S. Sedigh, and A. Miller. 2011. A Game-Theoretic Approach to Decision Support for Intelligent Water Distribution. In *Proceedings of the Hawaii International Conference on System Sciences HICSS '11* (Hawaii). IEEE, Washington, DC, USA, 1–10.

[133] Jing Lin, Sahra Sedigh, and Ann Miller. 2011. A Semantic Agent Framework for Cyber-Physical Systems. In *Semantic Agent Systems*, Janusz Kacprzyk, Atilla Elçi, Mamadou Tadiou Koné, and Mehmet A. Orgun (Eds.). Vol. 344. Springer Berlin Heidelberg, Berlin, Heidelberg, 189–213.

[134] Jingyong Liu and Lichen Zhang. 2011. Aspect-Oriented MDA Development Method for Non-Functional Properties of Cyber Physical Systems. In *Proceedings of the 2nd International Conference on Networking and Distributed Computing(ICNDC'11)* (Beijing). IEEE, Washington, DC, USA, 149–153.

[135] Yun Liu and Kishore S Trivedi. 2006. Survivability quantification: The analytical modeling approach. *International Journal of Performability Engineering* 2, 1 (2006), 29.

[136] T. A. Longstaff and Y. Y. Haimes. 2002. A Holistic Roadmap for Survivable infrastructure systems. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 32, 2 (2002), 260–268.

[137] Charles M. Macal and Michael J. North. 2005. Tutorial on agent-based modeling and simulation. In *Proceedings of the 37th Winter Simulation Conference WSC '05* (Orlando, Florida). IEEE, Washington, DC, USA, 2–15.

[138] Charles M. Macal and Michael J. North. 2006. Tutorial on agent-based modeling and simulation Part 2: How to model with agents. In *Proceedings of the 38th Winter Simulation Conference WSC '06*. IEEE, Washington, DC, USA, 73–83.

[139] Carlos Andrés Macana, Nicanor Quijano, and Eduardo Mojica-Nava. 2011. A survey on cyber physical energy systems and their applications on smart grids. In *Innovative Smart Grid Technologies (ISGT Latin America), 2011 IEEE PES Conference on*. IEEE, Washington, DC, USA, 1–7.

[140] M. Malhotra and K.S. Trivedi. 1995. Dependability modeling using Petri-nets. *Reliability, IEEE Transactions on* 44, 3 (Sep 1995), 428–440.

[141] Manish Malhotra and Kishor S Trivedi. 1994. Power-hierarchy of dependability-model types. *Reliability, IEEE Transactions on* 43, 3 (1994), 493–502.

[142] K. Marashi, S. Sedigh, and Ali R. Hurson. 2016. Quantification and Analysis of Interdependency in Cyber-Physical Systems. In *Proceedings of the 3rd International Workshop on Reliability and Security Aspects for Critical Infrastructure Protection (ReSA4CI '16)* (Toulouse, France). IEEE, Washington, DC, USA, 149–154.

[143] Koosha Marashi and Sahra Sedigh Sarvestani. 2014. Towards Comprehensive Modeling of Reliability for Smart Grids: Requirements and Challenges. In *Proceedings of the 15th IEEE International High Assurance Systems Engineering Symposium (HASE '14)* (Miami, FL). IEEE, Washington, DC, USA, 105–112.

[144] K Marashi, M Woodard, S Sedigh, and A Hurson. 2014. Quantitative reliability analysis for intelligent water distribution networks. *Transactions of the American Nuclear Society* (2014).

[145] MathWorks. 2017. SimEvents - Model and simulate discrete-event systems. http://www.mathworks.com/products/simevents/. Retrieved Feb. 21, 2017.

[146] MathWorks. 2017. Simulink - Simulation and Model-Based Design. http://www.mathworks.com/products/simulink/. Retrieved Feb. 21, 2017.

[147] Matrikon. 2017. OPC server from MatrikonOPC. http://www.matrikonopc.com/. Retrieved Feb. 21, 2017.

[148] Timothy McDaniels, Stephanie Chang, Krista Peterson, Joey Mikawoz, and Dorothy Reed. 2007. Empirical Framework for Characterizing Infrastructure Failure Interdependencies. *Journal of Infrastructure Systems* 13 (September 2007), 175–184.

[149] B. McMillin. 2009. Complexities of information security in Cyber-Physical Power Systems. In *IEEE/PES Power Systems Conference and Exposition PSCE '09*. IEEE, Washington, DC, USA, 1–2.

[150] John F. Meyer. 1980. On evaluating the performability of degradable computing systems. *Computers, IEEE Transactions on* 100, 8 (1980), 720–731.

[151] F. Milano. 2005. An Open Source Power System Analysis Toolbox. *IEEE Transactions on Power Systems* 20, 3 (August 2005), 1199–1206.

[152] L. Mkrtchyan, L. Podofillini, and V. N. Dang. 2015. Bayesian belief networks for human reliability analysis: A review of applications and gaps. *Reliability Engineering & System Safety* 139 (July 2015), 1–16. https://doi.org/10.1016/j.ress.2015.02.006

[153] Yuchang Mo, Liudong Xing, Farong Zhong, and Zhao Zhang. 2016. Reliability Evaluation of Network Systems with Dependent Propagated Failures Using Decision Diagrams. *IEEE Transactions on Dependable and Secure Computing* 13, 6 (Nov. 2016), 672–683. https://doi.org/10.1109/TDSC.2015.2433254

[154] Mohammad Modarres, Mark Kaminskiy, and Vasiliy Krivtsov. 1999. *Reliability Engineering and Risk Analysis: A Practical Guide*. CRC Press, Boca Raton, FL, USA.

[155] modelica.org. 2016. Modelica and the Modelica Association. https://www.modelica.org/. Retrieved November 30, 2016.

[156] Anitha Murugesan, Sanjai Rayadurgam, and Mats Heimdahl. 2013. Using models to address challenges in specifying requirements for medical cyber-physical systems. In *Fourth workshop on Medical Cyber-Physical Systems*. IEEE, Washington, DC, USA.

[157] Cen Nan and Giovanni Sansavini. 2017. A quantitative method for assessing resilience of interdependent infrastructures. *Reliability Engineering & System Safety* 157 (Jan. 2017), 35–53. https://doi.org/10.1016/j.ress.2016.08.013

[158] National Infrastructure Advisory Council. 2009. *Critical infrastructure resilience—final report and recommendations*. Technical report. DHS/NIAC. https://www.dhs.gov/xlibrary/assets/niac/niac_critical_infrastructure_resilience.pdf

[159] Thi Minh Luan Nguyen, Christophe Lecerf, and Ivan Lavallée. 2006. A space aware agent-based modeling process for the study of hierarchical complex systems. In *Annual ACM Symposium on Applied Computing SAC '06* (Orlando, Florida). ACM, New York, NY, USA, 126–127.

[160] Thomas Dyhre Nielsen and Finn Verner Jensen. 2009. *Bayesian networks and decision graphs*. Springer Science & Business Media, New York, NY, USA.

[161] NIST. 2006. 4009 National Information Assurance (IA) Glossary.

[162] NIST. 2013. *Strategic Vision and Business Drivers for 21st Century Cyber-Physical Systems.* Technical Report. NIST.

[163] nsnam. 2016. ns-3. https://www.nsnam.org/. Retrieved November 30, 2016.

[164] nsnam.sourceforge.org. 2011. The Network Simulator - ns-2. http://nsnam.sourceforge.net/wiki/index.php/Main_Page. Retrieved November 30, 2016.

[165] Henry Ohlef, William Binroth, and Roger Haboush. 1978. Statistical Model for a Failure Mode and Effects Analysis and Its Application to Computer Fault-Tracing. *Reliability, IEEE Transactions on* 27, 1 (1978), 16–22.

[166] Giustolisi Orazio. 2020. Water Distribution Network Reliability Assessment and Isolation Valve System. *Journal of Water Resources Planning and Management* 146, 1 (Jan. 2020), 04019064. https://doi.org/10.1061/(ASCE)WR.1943-5452.0001128

[167] Min Ouyang. 2014. Review on modeling and simulation of interdependent critical infrastructure systems. *Reliability engineering & System safety* 121 (2014), 43–60.

[168] Min Ouyang and Leonardo Dueñas-Osorio. 2012. Time-dependent resilience assessment and improvement of urban infrastructure systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 22, 3, Article 033122 (2012), 11 pages. https://doi.org/10.1063/1.4737204

[169] Z. Pan, Q. Xu, C. Chen, and X. Guan. 2016. NS3-MATLAB co-simulator for cyber-physical systems in smart grid. In *2016 35th Chinese Control Conference (CCC).* IEEE, Washington, DC, USA, 9831–9836.

[170] Behrooz Parhami. 1994. A multi-level view of dependable computing. *Computers & Electrical Engineering* 20, 4 (July 1994), 347–368. https://doi.org/10.1016/0045-7906(94)90048-5

[171] Marcus Pendleton, Richard Garcia-Lebron, Jin-Hee Cho, and Shouhuai Xu. 2016. A Survey on Systems Security Metrics. *ACM Comput. Surv.* 49, 4, Article 62 (Dec. 2016), 35 pages. https://doi.org/10.1145/3005714

[172] André Platzer and Jan-David Quesel. 2008. KeYmaera: A Hybrid Theorem Prover for Hybrid Systems (System Description). In *Automated Reasoning (Lecture Notes in Computer Science)*, Alessandro Armando, Peter Baumgartner, and Gilles Dowek (Eds.). Springer, Berlin, Heidelberg, 171–178. https://doi.org/10.1007/978-3-540-71070-7_15

[173] Claudius Ptolemaeus (Ed.). 2014. *System design, modeling, and simulation: using Ptolemy II* (1. ed., version 1.02 ed.). UC Berkeley EECS Dept, Berkeley, Calif.

[174] C. Queiroz, A. Mahmood, and Z. Tari. 2013. A Probabilistic Model to Predict the Survivability of SCADA Systems. *Industrial Informatics, IEEE Transactions on* 9, 4 (Nov 2013), 1975–1985. https://doi.org/10.1109/TII.2012.2231419

[175] Jose E. Ramirez-Marquez, Claudio M. Rocco, Kash Barker, and Jose Moronta. 2018. Quantifying the resilience of community structures in networks. *Reliability Engineering & System Safety* 169, Supplement C (Jan. 2018), 466–474. https://doi.org/10.1016/j.ress.2017.09.019

[176] Nageswara S. V. Rao, Chris Y. T. Ma, and David K. Y. Yau. 2011. On robustness of a class of Cyber-Physical Network Infrastructures. In *Proceedings of the 7th International Wireless Communications and Mobile Computing Conference(IWCMC'11)* (Istanbul). IEEE, Washington, DC, USA, 808–813.

[177] Adnan Rashid, Umair Siddique, and Sofiène Tahar. 2020. Formal Verification of Cyber-Physical Systems Using Theorem Proving. In *Formal Techniques for Safety-Critical Systems*, Osman Hasan and Frédéric Mallet (Eds.). Springer International Publishing, Cham, 3–18.

[178] Kaliappa Ravindran and Mohammad Rabby. 2011. Cyber-Physical Systems Based Modeling of Adaptation Intelligence in Network Systems. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics(SMC'11)* (Anchorage, AK). IEEE, Washington, DC, USA, 2737–2742.

[179] David Rehak, Pavel Senovsky, Martin Hromada, and Tomas Lovecek. 2019. Complex approach to assessing resilience of critical infrastructure elements. *International Journal of Critical Infrastructure Protection* 25 (June 2019), 125–138. https://doi.org/10.1016/j.ijcip.2019.03.003

[180] S. Rinaldi, J. Peerenboom, and T. Kelly. 2001. Identifying, understanding, and analyzing critical infrastructure interdependencies. *IEEE Control Systems Magazine* 11 (December 2001), 11–25.

[181] Ron Ross et al. 2013. Security and Privacy Controls for Federal Information Systems and Organizations. *NIST Special Publication* 800 (2013), 53–4.

[182] Kenneth C Rovers, Jan Kuper, Marcel D van de Burgwal, André BJ Kokkeler, and Gerard JM Smit. 2011. Mixed continuous/discrete time modelling with exact time adjustments. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International.* IEEE, Washington, DC, USA, 1111–1116.

[183] A.-E. Rugina, K. Kanoun, and M. Kaaniche. 2008. The ADAPT Tool: From AADL Architectural Models to Stochastic Petri Nets through Model Transformation. In *2008 Seventh European Dependable Computing Conference.* IEEE, Washington, DC, USA, 85–90. https://doi.org/10.1109/EDCC-7.2008.14

[184] Santiago Ruiz-Arenas, Imre Horvath, Ricardo Mejía-Gutiérrez, and Eliab Opiyo. 2014. Towards the Maintenance Principles of Cyber-Physical Systems. *Strojniški vestnik-Journal of Mechanical Engineering* 60, 12 (2014), 815–831.

[185] M. Rungger and P. Tabuada. 2016. A Notion of Robustness for Cyber-Physical Systems. *IEEE Trans. Automat. Control* 61, 8 (Aug 2016), 2108–2123. https://doi.org/10.1109/TAC.2015.2492438

[186] Florian Schupfer, Carna Radojicic, Joseph Wenninger, and Christoph Grimm. 2011. System Refinement Design Flow based on Semi-Symbolic Simulations. In *Proceedings of the AFRICON'11* (Livingstone). IEEE, Washington, DC, USA, 1–6.

[187] Lui Sha, Sathish Gopalakrishnan, Xue Liu, and Qixin Wang. 2008. Cyber-Physical Systems: A New Frontier. In *Proceedings of the 2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, SUTC '08*. IEEE Computer Society, Washington, DC, USA, 1–9. https://doi.org/10.1109/SUTC.2008.85

[188] Shey-Huei Sheu, Hsin-Nan Tsai, Fu-Kwun Wang, and Zhe George Zhang. 2015. An extended optimal replacement model for a deteriorating system with inspections. *Reliability Engineering & System Safety* 139 (July 2015), 33–49. https://doi.org/10.1016/j.ress.2015.01.014

[189] Jianhua Shi, Jiafu Wan, Hehua Yan, and Hui Suo. 2011. A survey of cyber-physical systems. In *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*. IEEE, Washington, DC, USA, 1–6.

[190] Daniel P Siewiorek, John J Hudak, Byung-Hoon Suh, and Zary Segal. 1993. Development of a benchmark to measure system robustness. In *Fault-Tolerant Computing, 1993. FTCS-23. Digest of Papers., The Twenty-Third International Symposium on*. IEEE, Washington, DC, USA, 88–97.

[191] Marjan Sirjani, Edward A. Lee, and Ehsan Khamespanah. 2020. Verification of Cyberphysical Systems. *Mathematics* 8, 7 (July 2020), 1068. https://doi.org/10.3390/math8071068 Multidisciplinary Digital Publishing Institute.

[192] R.M. Smith, K.S. Trivedi, and A.V. Ramesh. 1988. Performability analysis: measures, an algorithm, and a case study. *Computers, IEEE Transactions on* 37, 4 (Apr 1988), 406–417.

[193] Alexandru Stefanov and Chen-Ching Liu. 2012. ICT modeling for integrated simulation of cyber-physical power systems. In *Innovative Smart Grid Technologies (ISGT Europe), 2012 3rd IEEE PES International Conference and Exhibition on*. IEEE, Washington, DC, USA, 1–8. https://doi.org/10.1109/ISGTEurope.2012.6465730

[194] Yan Sun, Bruce McMillin, Xiaoqing (Frank) Liu, and David Cape. 2007. Verifying Noninterference in a Cyber-Physical System The Advanced Electric Power Grid. In *Proceedings of the Seventh International Conference on Quality Software, QSIC '07*. IEEE Computer Society, Washington, DC, USA, 363–369.

[195] Yoshihiko Susuki, T. John Koo, Hiroaki Ebina, Takuya Yamazaki, Takashi Ochi, Takuji Uemura, and Takashi Hikihara. 2012. A Hybrid System Approach to the Analysis and Design of Power Grid Dynamic Performance. *Proc. IEEE* 100 (January 2012), 225–239. Issue 1.

[196] Janos Sztipanovits. 2007. Composition of Cyber-Physical Systems. In *Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems, ECBS '07*. IEEE Computer Society, Washington, DC, USA, 3–6. https://doi.org/10.1109/ECBS.2007.25

[197] Ying Tan, Steve Goddard, and Lance C. Pérez. 2008. A prototype architecture for cyber-physical systems. *ACM SIGBED Review* 5, 1 (2008), 1–2. https://doi.org/10.1145/1366283.1366309

[198] Ying Tan, Mehmet C. Vuran, and Steve Goddard. 2009. Spatio-Temporal Event Model for Cyber-Physical Systems. In *Proceedings of the 29th IEEE International Conference on Distributed Computing Systems Workshops, ICDCS Workshops '09*. IEEE, Washington, DC, USA, 44 –50. https://doi.org/10.1109/ICDCSW.2009.82

[199] Chee-Wooi Ten, Chen-Ching Liu, and M. Govindarasu. 2008. Anomaly extraction and correlations for power infrastructure cyber systems. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics SMC '08* (Singapore). IEEE, Washington, DC, USA, 7–12.

[200] Y. Van Tendeloo and H. Vangheluwe. 2017. The Modelverse: A tool for Multi-Paradigm Modelling and simulation. In *2017 Winter Simulation Conference (WSC)*. IEEE, Las Vegas, NV, USA, 944–955. https://doi.org/10.1109/WSC.2017.8247845

[201] Scott Thacker, Jim W. Hall, and Raghav Pant. 2018. Preserving Key Topological and Structural Features in the Synthesis of Multilevel Electricity Networks for Modeling of Resilience and Risk. *Journal of Infrastructure Systems* 24, 1 (March 2018), 04017043. https://doi.org/10.1061/(ASCE)IS.1943-555X.0000404

[202] The International Telegraph and Telephone Consultative Committee. 2008. ITU-T Recommendation E.800. Quality of Telecommunication Services: Concepts, Models, Objectives and Dependability Planning. Terms and Definitions Related to the Quality of Telecommunication Services.

[203] Lothar Thiele and Pratyush Kumar. 2015. Can real-time systems be chaotic?. In *2015 International Conference on Embedded Software (EMSOFT)*. IEEE, Amsterdam, Netherlands, 21–30. https://doi.org/10.1109/EMSOFT.2015.7318256

[204] Iris Tien and Armen Der Kiureghian. 2016. Algorithms for Bayesian network modeling and reliability assessment of infrastructure systems. *Reliability Engineering & System Safety* 156 (Dec. 2016), 134–147. https://doi.org/10.1016/j.ress.2016.07.022

[205] Stavros Tripakis and Costas Courcoubetis. 1996. Extending Promela and Spin for real time. In *International Workshop on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, Cham, 329–348. http://link.springer.com/chapter/10.1007/3-540-61042-1_53

[206] Kisho S. Trivedi and Robin Sahner. 2009. SHARPE at the age of twenty two. *ACM SIGMETRICS Performance Evaluation Review* 36, 4 (March 2009), 52–57. https://doi.org/10.1145/1530873.1530884

[207] M. Ulieru. 2007. Design for Resilience of Networked Critical Infrastructures. In *Inaugural IEEE-IES Digital EcoSystems and Technologies Conference DEST '07*. IEEE, Washington, DC, USA, 540–545.

[208] US EPA. 2016. EPANET | Water Research | US EPA. https://www.epa.gov/water-research/epanet. Retrieved December 1, 2016.

[209] I. B. Utne, P. Hokstad, and J. Vatn. 2011. A method for risk modeling of interdependencies in critical infrastructures. *Reliability Engineering & System Safety* 96, 6 (June 2011), 671–678. https://doi.org/10.1016/j.ress.2010.12.006

[210] Jørn Vatn, Per Hokstad, and Ingrid Bouwer Utne. 2012. Defining Concepts and Categorizing Interdependencies. In *Risk and Interdependencies in Critical Infrastructures*, Per Hokstad, Ingrid B. Utne, and Jørn Vatn (Eds.). Springer London, London, UK, 13–22. http://link.springer.com.libproxy.mst.edu/chapter/10.1007/978-1-4471-4661-2_2

[211] Trivik Verma, Wendy Ellens, and Robert E. Kooij. 2015. Context-independent centrality measures underestimate the vulnerability of power grids. *International Journal of Critical Infrastructures 7* 11, 1 (2015), 62–81. http://www.inderscienceonline.com/doi/abs/10.1504/IJCIS.2015.067398

[212] Yunbo Wang, Mehmet C. Vuran, and Steve Goddard. 2008. Cyber-physical systems in industrial process control. *ACM SIGBED Review* 5, 1 (2008), 1–2. https://doi.org/10.1145/1366283.1366295

[213] W. Wolf. 2009. Cyber-physical Systems. *Computer* 42, 3 (March 2009), 88–89. https://doi.org/10.1109/MC.2009.81

[214] H. Woo, J. Yi, J.C. Browne, A.K. Mok, E. Atkins, and F Xie. 2008. Design and Development Methodology for Resilient Cyber-Physical Systems. In *28th International Conference on Distributed Computing Systems Workshops ICDCS '08*. IEEE, Washington, DC, USA, 525–528.

[215] Anthony D. Wood and John A. Stankovic. 2008. Human in the loop: distributed data streams for immersive cyber-physical systems. *ACM SIGBED Review* 5, 1, Article 20 (2008), 2 pages.

[216] Mark Woodard, Koosha Marashi, and Sahra Sedigh Sarvestani. 2017. Survivability Evaluation and Importance Analysis for Complex Networked Systems. *IEEE Transactions on Network Science and Engineering* (2017). Submitted.

[217] L. Wu and G. Kaiser. 2013. FARE: A framework for benchmarking reliability of cyber-physical systems. In *Systems, Applications and Technology Conference (LISAT), 2013 IEEE Long Island*. IEEE, Washington, DC, USA, 1–6. https://doi.org/10.1109/LISAT.2013.6578226

[218] L. Xie and M.D. Ilic. 2008. Module-Based Modeling of Cyber-Physical Power Systems. In *28th International Conference on Distributed Computing Systems Workshops ICDCS '08*. IEEE, Washington, DC, USA, 513–518.

[219] Le Xie and Marija D Ilic. 2008. Module-based modeling of cyber-physical power systems. In *Distributed Computing Systems Workshops, 2008. ICDCS'08. 28th International Conference on*. IEEE, Washington, DC, USA, 513–518.

[220] Chengchao Xu and Ian C. Goulter. 1998. Probabilistic model for water distribution reliability. *Journal of Water Resource Planning and Management* 124 (July/August 1998), 218–228.

[221] Osman Yagan, Dajun Qian, Junshan Zhang, and Douglas Cochran. 2012. Optimal allocation of interconnecting links in cyber-physical systems: Interdependence, cascading failures, and robustness. *Parallel and Distributed Systems, IEEE Transactions on* 23, 9 (2012), 1708–1720.

[222] Rongfei Zeng, Yixin Jiang, Chuang Lin, and Xuemin Shen. 2012. Dependability analysis of control center networks in smart grid using stochastic petri nets. *Parallel and Distributed Systems, IEEE Transactions on* 23, 9 (2012), 1721–1730.

[223] Lichen Zhang. 2011. Aspect-Oriented MDA Approach for Non-Functional Properties of Distributed Cyber Physical Systems. In *Proceedings of the 10th International Symposium on Distributed Computing and Applications to Business, Engineering and Science(DCABES'11)* (Wuxi, Jiangsu). IEEE, Washington, DC, USA, 284–288.

[224] Lichen Zhang. 2011. Formal Support for Cyber Physical System Specification Using Aspect-Oriented Approach. In *Proceedings of the 10th International Symposium on Distributed Computing and Applications to Business, Engineering and Science(DCABES'11)* (Wuxi, Jiangsu). IEEE, Washington, DC, USA, 31–35.

[225] Lichen Zhang and Jifeng He. 2011. Aspect-Oriented QoS Specification for Cyber-Physical Systems. In *Proceedings of the 5th International Conference on Convergence and Hybrid Information Technology(ICHIT'11)* (Daejeon). IEEE, Washington, DC, USA, 399–406.

[226] Le-Jun Zhang, Wei Wang, Lin Guo, Yang Wu, and Yong-Tian Yang. 2007. A survivability quantitative analysis model for network system based on attack graph. In *Machine Learning and Cybernetics, 2007 International Conference on*, Vol. 6. IEEE, IEEE, Washington, DC, USA, 3211–3216.

[227] Zuyuan Zhang, Wei An, and Fangming Shao. 2016. Cascading Failures on Reliability in Cyber-Physical System. *IEEE Transactions on Reliability* 65, 4 (Dec. 2016), 1745–1754. https://doi.org/10.1109/TR.2016.2606125

[228] Yang Zhao and Kristin Y. Rozier. 2014. Formal specification and verification of a coordination protocol for an automated air traffic control system. *Science of Computer Programming* 96 (2014), 337–353.

[229] Kalman Žiha. 2000. Redundancy and robustness of systems of events. *Probabilistic engineering mechanics* 15, 4 (2000), 347–357.

Received February 2017