



Probabilistic Inference of Fault Condition of Cyber-Physical Systems Under Uncertainty

Xin Tao , Jinzhi Lu , Dejiu Chen , *Senior Member, IEEE*, and Martin Törngren , *Senior Member, IEEE*

Abstract—Cyber-physical systems (CPS) are paving new ground with increasing levels of automation and usage in applications with complex environments, posing greater challenges in terms of safety and reliability. The increasing complexity of CPS environments, tasks, and systems leads to more uncertainties. Unless properly managed, these uncertainties may lead to false detection of real fault condition of a system, which in turn may affect decision making and potentially cause fatal consequences. In order to implement safety-critical missions, such as autonomous driving, it is essential to develop a reliable monitoring and assessment service dealing with the complexity and uncertainty issues. In this article, we propose a fault detection function based on Bayesian inference, which combines empirical knowledge with information of the specific system. By considering uncertainties as possible causes for false detection, various uncertainties during the detection process are analyzed, and the ways to quantify and propagate them are explored. As a result, probabilistic inference is achieved for distinguishing system faults from uncertainties, which contributes to more reliable detection results regarding system faults under dynamically changing environments. A case study on an microelectro mechanical system (MEMS) accelerometer is conducted and the result shows that the fault detection function effectively distinguishes system faults and uncertainties arising from the environment.

Index Terms—Bayesian inference (BI), cyber-physical systems (CPS), fault detection, monitoring and assessment service (MAS), uncertainty.

I. INTRODUCTION

CYBER-PHYSICAL systems (CPS) involve computation, communication, sensing, and actuation, as well as physical processes [1]. In 2017, a framework for CPS was released by the National Institute of Standards and Technology (NIST), where monitoring, anomaly detection, and self-diagnostics are listed as fundamental and important functionalities for CPS operations [1]. In CPS, monitoring and assessment of system fault conditions are critical for decision making, thus influencing the safety and reliability of mission-critical CPS, such as automotive systems and smart grids [2], [3]. In [4], a monitoring and assessment service (MAS) was proposed for monitoring the conditions of a system and its components. The term MAS is

used in this article to refer to the MAS of system fault conditions in CPS. When MAS fails to detect the real fault conditions, a failure of this service occurs, which potentially results into other system failure and influences system safety and reliability. In this article, the hypothesized causes leading to such MAS service failure are called system faults, and fault condition refers to the situation that whether system faults exist [5].

In real situations, due to the increasing complexity of CPS and operational environment, there is increasing uncertainty, which causes rising concerns and issues of fault detection in MAS. The definition of uncertainty depends on the specific context. In this article, the definition of uncertainty is restricted to refer to possible causes of false detection of system faults. There are various uncertainties that influence fault detection results, for example, environment noise, sensing system failure, lacking of model knowledge, etc. These uncertainties may lead to false detection of the real fault conditions, potentially posing risks to personal safety and finances. For example, since the parameters of some sensor materials, like semiconductors vary when the ambient temperature exceeds the threshold, the accuracy of sensors deteriorates, which may lead to biased data and false detection of faults. Such false fault detection further influences the decision-making process. An implication is that erroneous decision could be made to perform offline maintenance based on false fault detection, with potentially large financial loss for a large product line. In these situations, the reliability of the detection result is of great importance in the CPS development. When dealing with these uncertainties, it is critical, albeit challenging, to find an efficient way to evaluate their impact on fault detection.

Thus, in this article, we take on the problem of *evaluating the impact of various uncertainties on fault detection of CPS*. This article's contribution of tackling the problem is threefold. First, we formulate and specify the problem regarding the impact of uncertainty on fault detection in CPS. Second, we propose an uncertainty analysis framework to evaluate and quantify uncertainties that potentially result in false fault detections. Third, based on the proposed uncertainty analysis framework, we develop a fault detection function in MAS for CPS, which provides probabilistic inference regarding system fault condition based on Bayesian inference (BI).

The rest of the article is organized as follows. In Section II, research design is presented from two aspects: the state-of-the-art and research methodology. In Section III, the design of an uncertainty analysis framework and a fault detection function is presented. In Section IV, a case study based on an micro

Manuscript received March 29, 2019; revised September 27, 2019; accepted December 31, 2019. Date of publication January 24, 2020; date of current version September 2, 2020. This work of X. Tao was supported by China Scholarship Council (CSC). (Corresponding author: Xin Tao.)

X. Tao, D. Chen, and M. Törngren are with the Department of Machine Design, KTH Royal Institute of Technology, Stockholm SE-100 44, Sweden (e-mail: taoxin@kth.se; chen@md.kth.se; martint@kth.se).

J. Lu is with the École Polytechnique Fédérale de Lausanne, Lausanne 1015, Switzerland (e-mail: jinzhi.lu@epfl.ch).

Digital Object Identifier 10.1109/JSYST.2020.2965400

electro mechanical system (MEMS) accelerometer is conducted to validate the authenticity of the proposed problem, the usability of the proposed function, and its performance. Finally, the conclusion and future work are discussed in Section V.

II. RESEARCH DESIGN

This section consists of two parts: state-of-the-art and research methodology. The state of the art contributes to the motivations for developing a fault detection function in MAS considering various uncertainties. The research methodology section illustrates the procedures and techniques used in this article.

A. State of the Art

Over the past few years, researchers have proposed different approaches for fault detection in various engineering fields, such as automotive [6], smart grids [7] and aviation [8]. In general, traditional fault detection methods may be classified into two major groups: model-free methods and model-based methods [9]. Model-free methods typically include physical redundancy, limit checking, and so on, whereas model-based methods utilize an explicit mathematical model of the monitored plant. Except for the model of physical plant, fault model is also typical for fault detection. Such a model is built based on a mapping rule between the known failure modes in a system and the available symptoms, often captured in a dependence matrix [6]. Many conventional fault detection methods are based on this type of model [10], [11], which is effective when accurate fault models are developed. However, complex CPS makes it difficult to create such an accurate model. With the emergence of technologies like machine learning, big data, and cloud computing, data-driven methods are playing increasingly important role in fault detection. Specifically, anomaly detection find patterns in data that do not conform to expected behavior, and is applied to various domains, like fault detection and intrusion detection [12], [13]. Various machine learning algorithms have been adapted for fault detection, such as decision trees [14], artificial neural networks, and support vector machines [15]. However, the availability, accessibility, quality, and volume of the training data are critical. Especially, in the development phase of MAS, field data regarding system faults are lacking and empirical data are partial and no longer qualified due to uncertainties arising from specific systems and complex operational environments.

The research of uncertainty in CPS has been conducted in recent years. The need for comprehensive uncertainty treatment in CPS has been recognized [1], and the taxonomy, characterization, and propagation of uncertainty are under active exploration [16]–[18]. In the guidelines released by NIST, the measurement uncertainties were grouped into two categories, i.e., ones evaluated by statistical methods and ones evaluated using all the relevant information available [19]. These two categories are also referred to as aleatory uncertainties and epistemic uncertainties [20]. Aleatory uncertainty refers to the inherent randomness correlated to the physical system or environment, whereas epistemic uncertainty is mainly derived from the lack of knowledge or information (domain unfamiliarity, limited experience, etc.). This classification method is also practicable

regarding CPS. In fault detection, aleatory uncertainties mainly arise from noise, temperature, system inputs, and the nature of sensing and measurement equipment, whereas epistemic uncertainties mainly include partial model, poor data quality, experiment assumptions, etc. Different analysis methods for uncertainty evaluation, including probability bound analysis, imprecise probability, evidence theory, and possibility theory, are explored in [21] and [22]. Despite these efforts, uncertainty modeling, quantification, and propagation in CPS remain challenging due to complex operational environments and limited knowledge of the system.

Regarding uncertainty treatment in MAS, Bayesian approach is a typical statistical method that effectively deals with various uncertainty problems and is increasingly utilized in fault diagnosis [23]. In [24], board-level fault diagnosis was conducted using BI for pattern analysis and classification. A Bayesian fault detection algorithm was proposed in [25] for detecting an abrupt latent fault in a sensor. In [26], Bayesian networks and causal modeling were used in decision making under risk-related uncertainty. In these works, the detection accuracy was improved by integrating information of offline fault-insertion tests, whereas uncertainty-related temporal characteristics of system variables and processes are less considered. In [27], a Bayesian network approach was presented for anomaly detection by learning the causal relations between cyber and physical variables as well as their temporal correlations from unlabeled data. In [28], dynamic Bayesian networks are used to represent the temporal characteristics of the normal process regarding anomaly detection. Except for Bayesian approaches, the hidden Markov model (HMM), with good performance in time-varying dynamic systems, also gains increasing attention. In [29], an HMM-based algorithm was developed for online fault detection with partial and imperfect tests. These test uncertainties are handled to find the best inference of fault conditions and to identify the dynamic changes in fault conditions. A simulation-based approach was introduced in [30], using discrete-time Markov chains and a probabilistic model accommodating a diverse set of parameter range distributions for software architecture evaluation. These methods based on Markov chains had a common assumption that the evolution of system failure modes could be approximated by a parametric random process with Gaussian distribution.

The literature reviews reveal that the systematic uncertainty reasoning in a quantitative way for fault detection in CPS is still lacking. It is commonly the case that only certain types of uncertainties, like signal noise, were taken into consideration for improving fault detection performance. In this article, a more systematic uncertainty analysis framework including uncertainty quantification and propagation during the fault detection process is proposed. This framework is further utilized to design a fault detection function in MAS based on probabilistic inference using BI.

B. Research Methodology

1) Systems Thinking: Systems thinking is used to improve the capability of identifying and understanding systems, predicting their behaviors, and devising modifications to them in

TABLE I
MEASUREMENT AND METRICS TO EVALUATE THE PROPOSED FUNCTION

Measurement	Metrics	Questions
Authenticity of the problem statement	Impact of uncertainty on fault detection	Does uncertainty also result in features that indicate certain kind of fault?
Usability of the proposed fault detection function	Parameter values related to uncertainties	Can these parameters be acquired by the proposed methods?
Performance of the proposed fault detection function	Fault detection accuracy	Can the proposed function increase fault detection accuracy?

order to produce desired effects [31]. In this article, we conduct systems thinking with the following five steps.

- a) Define the problem: Through an explicit problem definition, the boundary, goal, and compositions of the system are clarified. In our case, the system boundary is the MAS of CPS and the goal is to develop a fault detection function considering heterogeneous uncertainties in complex CPS. The system is composed of CPS modules including MAS, system faults, and uncertainties. In order to analyze the connections between different compositions, a system model within the boundary is established, as shown in Section III-A.
- b) Design measurements and metrics: In Section I, three contributions of this article are listed. In order to evaluate the validity of these contributions, three measurements are designed. First, the first measurement is designed for the first contribution—formulate and specify the problem—the second and the third measurements are both designed for the third contribution—develop a fault detection function in MAS. No measurements are designed for the second contribution because the aim of the second contribution was not to assess but to propose an uncertainty analysis framework. However, this contribution is necessary to further support the design of a new fault detection function. Second, the metrics are proposed to enable the quantitative measurements. Lastly, example questions are presented to help to specify relevant metrics. These metrics and questions are used in an accelerometer use case that is presented in Section IV.
- c) Implement measurements: Based on the defined measurements and metrics, methods for calculating the metrics and tools for implementing the methods are analyzed, assessed, and selected with the target of problem solving. To realize fault detection function in MAS, different signal processing methods are used to extract features containing system fault information, and features are further filtered and chosen to represent system fault information. For example, for the first measurement and metrics in Table I, the feature extraction method is based on fast Fourier transform (FFT) and the tool is the signal processing toolbox of MATLAB.
- d) Formalize function: Having selected basic methods and tools, methods regarding to the specific problems are proposed, and a complete fault detection function integrating

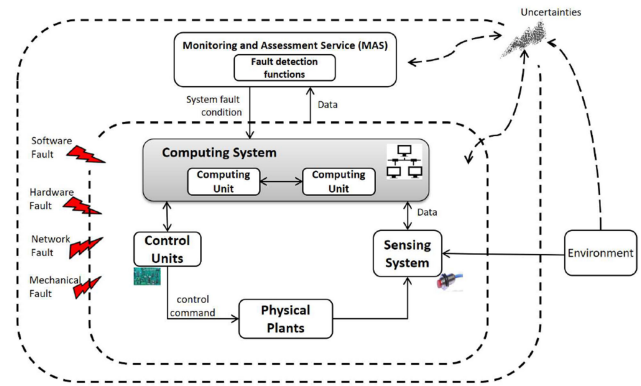


Fig. 1. Abstract model of CPS showing how fault detection function is implemented in CPS.

these methods is framed and formalized in an integrated way.

- e) Testing using a case study: In order to promote the understanding and to validate the proposed function, specialized user cases are tested using simulation with an established environment.

2) *Case Study*: A case study is an efficient way to tackle complex problems in real-world applications in a more concrete way. In this article, a case study is conducted to validate our function from several facets, shown as the measurements in Table I. By comparing the proposed function with a conventional function, the performance of the proposed function is clarified.

III. DESIGN OF AN UNCERTAINTY ANALYSIS FRAMEWORK AND A FAULT DETECTION FUNCTION

A. Problem Statement

An abstract model of CPS is presented in Fig. 1, which is based on the framework of CPS released by NIST [1]. This model shows how fault detection function is implemented in CPS. In this figure, there are two bounded blocks with dashed rectangle. The inner one includes computing system, communication system, sensing system, control units, and physical plants, which are interconnected and form a CPS. The outer one separates the environment and the CPS system. As an independent embedded software service, MAS lies in between of the two blocks, illustrating that such a service functions independently and may exist in different subsystems. The boundaries are dashed since systems at different levels may interact flexibly.

Fault detection function, as part of MAS, is used to detect faults arising from software, hardware, network, or physical plants. In general, MAS receives data from different subsystems, executes fault detection functions, and outputs system fault condition to the computing system for further decision making. Various uncertainties, either from the environment or from the system itself, influence the system from various aspects in return. Specifically, they will influence the fault detection accuracy, thus threatening the normal functioning of MAS, as well as the whole system. The uncertainty sources and their influence are very complex, which need to be clarified for a specific situation.

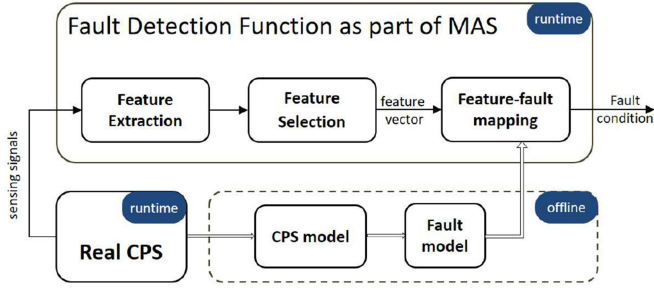


Fig. 2. Conventional fault detection function as part of MAS.

Dashed lines are used here to represent the interactions between uncertainties, environment, and the CPS system, whereas the solid lines represent wire or wireless transmission of signals and data flows.

A conventional fault detection function targeting at CPS is shown in Fig. 2. More details about the conventional fault detection function are presented in [6] and [32]. With a specified real CPS, the CPS model and the fault model of the system are derived. During runtime of MAS, sensing signals containing information of system faults are imported into MAS. Then, fault detection is performed, mainly with three steps, i.e., feature extraction, feature selection, and feature-fault mapping. First, feature extraction algorithms are applied to signal inputs and extract the features related to the system faults. Main feature extraction methods include time-domain analysis, frequency-domain analysis, and time-frequency analysis. Then, the features that indicate system faults are selected. There are lots of research on feature selection [33]. Finally, feature-fault mapping is performed based on the established fault model, and the fault condition of the system is exported. In this function, only when the CPS system and the corresponding system fault model are specified, the feature extraction algorithms and the selected features can be specified.

In this fault detection function, the feature-fault mapping relation is basically derived from the fault model of the system, which is developed offline. In a complex CPS that is operating in a dynamically changing operational environment, various uncertainties may affect the fault detection accuracy. One of the impacts is that the features regarded as symptoms of system faults are potentially caused by uncertainties, like environment noise, sensing device failure, and so on. In this situation, the conventional fault detection function based on empirical fault models may provide false detection results.

In the following section, we analyze the uncertainties involved in the fault detection function shown in Fig. 2.

B. Uncertainty Analysis Framework

1) *Basic Notations*: Later contents in this article involve mathematical descriptions. Here, basic notations of important variables of a specific system in a specific operational scenario are provided as follows.

- 1) System faults Fa : There are I kinds of system faults fa_i in total, i.e., $Fa = \{fa_1, \dots, fa_I : I \in \mathbb{N}\}$.

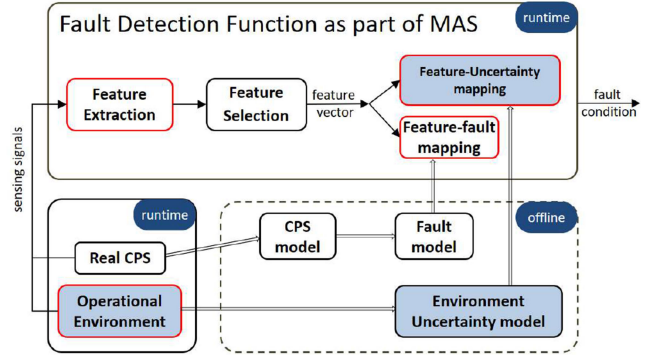


Fig. 3. Uncertainty analysis of conventional fault detection function in MAS.

- 2) Uncertainties U : In total, K kinds of uncertainties u_k are considered, i.e., $U = \{u_1, \dots, u_K : K \in \mathbb{N}\}$.
- 3) Signal features Fe : For a specific CPS, according to the fault model of the system, a total number of J features fe_j are extracted, i.e., $Fe = \{fe_1, \dots, fe_J : J \in \mathbb{N}\}$.

Each of these system variables is a set containing multiple elements. Fundamental assumptions are made in this article that 1) different system faults are independently distributed, and 2) system faults and uncertainties are independently distributed. As a result, the propagation of different faults and the propagation of faults and uncertainties are not considered in this article. Instead, this article explores the propagation of different uncertainties.

2) *Uncertainty Analysis and Mathematical Definition*: In this section, an uncertainty analysis framework, as shown in Fig. 3, is proposed for identifying uncertainties involved in the fault detection function presented in Fig. 2. In this framework, three blue boxes are added on top of Fig. 2. First, the operational environment of CPS is considered. Second, the environment uncertainty model is derived offline for the operational environment. Third, a feature-uncertainty mapping is established. Therefore, when the features are extracted and selected, they will not only be mapped to system faults but also to uncertainties. Under this framework, four kinds of uncertainties are identified and considered. The related sources of these uncertainties are marked as boxes with red outline.

- 1) Environment uncertainty: Environment uncertainty refers to the uncertainty of the operational environment of CPS. Such uncertainty is domain specific. For example, for an automated driving system, unexpected occurrence of bad road condition, weather condition, or lighting conditions may affect sensing signals of the system. Environment noise is also a typical uncertain environment factor. The modeling of environment uncertainty can be specified mathematically when the CPS system and the operational environment are specified.
- 2) The uncertainty of the feature extraction algorithms to detect the features: This type of uncertainty, denoted as m_j , is defined as the probability that the signal feature fe_j can be detected by a specific feature detection algorithm.
- 3) The uncertainty of the causal relation between the system fault and the generation of corresponding features: A

system fault does not necessarily result in the generation of a signal feature. For every system fault fa_i , we assume that the parameter space of fa_i is S^j and a specific feature fe_j is generated in a subspace $S^i_{fe_j}$. The parameter space here refers to the possible parameter values of variables. The probability that the feature fe_j is generated when the fault fa_i exists is denoted as $p^*(fe_j|fa_i)$ and defined as (1). p^* is used to distinguish from $p(fe_j|fa_i)$, which represents the probability that fe_j is detected when fa_i exists

$$p^*(fe_j|fa_i) = S^i_{fe_j}/S^i. \quad (1)$$

To propagate the uncertainties described in b) and c), the probability that feature fe_j is detected when system fault fa_i exists is defined as follows:

$$p(fe_j|fa_i) = S^i_{fe_j} \cdot m_j/S^i. \quad (2)$$

- 4) The uncertainty of the causal relation between environment uncertainties and the generation of corresponding features: The detected features result from system fault or environment uncertainties like noise, temperature, and so on. These environment uncertainties belong to the uncertainty set U . The parameter space of environment uncertainty u_k is defined as S^{*k} and a specific feature fe_j is generated in a subspace $S^{*k}_{fe_j}$. The probability that feature fe_j is generated when environment uncertainty u_k exists is defined as (3). The definition of parameter space and the reason of using p^* instead of p are similar as described in c)

$$p^*(fe_j|u_k) = S^{*k}_{fe_j}/S^{*k}. \quad (3)$$

To propagate uncertainties described in b) and d), the probability that feature fe_j is detected when environment uncertainty u_k exists is defined as follows:

$$p(fe_j|u_k) = S^{*k}_{fe_j} \cdot m_j/S^{*k}. \quad (4)$$

C. Probabilistic Inference for Fault Detection Based on BI

This section deals with the identified uncertainties in the last section by proposing probabilistic inference of fault condition based on BI.

1) *Probability Inference Based on BI*: In this section, we assume that 1) the CPS model, the fault model, and the uncertainty model are specified, and 2) the feature to be extracted is chosen and corresponds to a certain system fault. After importing sensing signals into MAS, if the chosen feature is detected, there are following four possible causes:

- 1) Cause A: system faults;
- 2) Cause B: uncertainties;
- 3) Cause C: both system faults and uncertainties;
- 4) Cause D: neither system faults nor uncertainties.

Here, the event that the detected feature is caused by neither system faults nor uncertainties is denoted by *nor*. On the condition that feature fe_j is detected, the probabilities of the four

Algorithm 1: Decision Process of the Output of the Fault Detection Function.

```

01: Calculate
     $p(fa_i|fe_j), p(u_k|fe_j), p((fa_i, u_k)|fe_j), p(nor|fe_j)$ 
02: for all  $j, k \in \mathbb{N}, j \leq J, k \leq K, \forall i \in \mathbb{N}, i \leq I$ 
03:   if  $p(fa_i|fe_j) > p(u_k|fe_j)$  or
     $p((fa_i, u_k)|fe_j) > p(u_k|fe_j)$ 
04:     and  $p(nor|fe_j) < \min(p(fa_i|fe_j), p(u_k|fe_j),$ 
     $p((fa_i, u_k)|fe_j))$ 
05:   then  $p(fa|fe_j) = 1$ 
06:   else  $p(fa|fe_j) = 0$ 
07: end for

```

causes are described using BI in the following equation:

$$p(fa_i|fe_j) = p(fe_j|fa_i)p(fa_i)/p(fe_j)$$

$$p(u_k|fe_j) = p(fe_j|u_k)p(u_k)/p(fe_j)$$

$$p((fa_i, u_k)|fe_j) = p(fe_j|(fa_i, u_k))p(fa_i, u_k)/p(fe_j)$$

$$p(nor|fe_j) = p(fe_j|nor)p(nor)/p(fe_j)$$

$$(i, j, k \in \mathbb{N}; i \leq I, j \leq J, k \leq K) \quad (5)$$

where $p(fa_i|fe_j)$ refers to the posterior probability that feature fe_j is caused by system fault fa_i ; $p(u_k|fe_j)$ refers to the posterior probability that feature fe_j is caused by uncertainty u_k ; $p((fa_i, u_k)|fe_j)$ refers to the posterior probability that feature fe_j is caused by both system fault fa_i and uncertainty u_k ; and $p(nor|fe_j)$ refers to the posterior probability that feature fe_j is caused by neither system faults nor uncertainties. $p(fe_j|fa_i)$, $p(fe_j|u_k)$, $p(fe_j|(fa_i, u_k))$, and $p(fe_j|nor)$ refer to the probabilities of detecting the feature fe_j with system fault fa_i , uncertainty u_k , both of them, and neither of them, respectively. These probabilities are usually called likelihoods. $p(fa_i)$, $p(u_k)$, $p(fa_i, u_k)$, and $p(nor)$ refer to the prior probabilities of system faults, uncertainties, both of them, and neither of them, respectively. $p(fe_j)$ refers to the marginal probability that feature fe_j is detected with all possible causes.

When feature fe_j is detected, the posterior probabilities of the four causes are obtained with (5). These posterior probabilities are further used to decide the output of the fault detection function. The decision process is shown in Algorithm 1. For each system fault, for all the corresponding features and uncertainties involved, if the following two conditions are satisfied:

- 1) the detected feature is more likely to be caused by system faults (Cause A) or both system faults and uncertainties (Cause C) than only uncertainties (Cause B);
- 2) the detected feature is least likely to be caused by neither system faults or uncertainties (Cause D)

then the output is “system fault,” denoted by $p(fa|fe_j) = 1$ in Algorithm 1. Otherwise, the output is “normal,” denoted by $p(fa|fe_j) = 0$ in Algorithm 1.

2) *Obtaining Probabilistic Parameters in BI*: In the previous section, probabilistic inference based on BI is introduced to calculate the posterior probabilities of all the possible causes

of a detected feature. In order to perform the calculation, the likelihood, prior, and marginal probabilities of BI in (5) need to be obtained first. In this section, the methods of obtaining these parameter values are proposed based on the uncertainty analysis framework in Section III-B.

During the development phase of MAS, the value of $p(fa_i)$ is obtained with empirical knowledge. There are various standards and handbooks about the failure rate of components [34]–[36]. As analyzed in Section III-B, uncertainties may arise from the system itself or the environment. For uncertainties from the environment, the value of $p(u_k)$ could be obtained by other monitoring services or empirical knowledge. For uncertainties from the system, simulations or refined system modeling could be used for obtaining the value of $p(u_k)$. For example, the second kind of uncertainty identified in Section III-B involves the performance of the feature extraction algorithm, which could be explored with simulations. With the assumption that system faults and uncertainties are independent, $p(fa_i, u_k)$ is obtained with the following equation:

$$p(fa_i, u_k) = p(fa_i) \cdot p(u_k). \quad (6)$$

Since these four causes cover all the possible conditions of the system, the sum of the prior probabilities of these four causes satisfies the following equation:

$$\sum_i p(fa_i) + \sum_k p(u_k) + \sum_{i,k} p(fa_i, u_k) + p(nor) = 1. \quad (7)$$

The value of $p(nor)$ is obtained by substituting the value of $p(fa_i)$, $p(u_k)$, and $p(fa_i, u_k)$ into (7).

In order to obtain the value of the likelihoods, simulations are used, which is discussed in the following section. According to the law of large numbers [37], when the number of simulations is large enough, the proportion of the frequency is an infinite approximation of the probability. So by generating a large number of simulations with system faults and adding up the number of simulations where a feature is successfully detected, the approximation of the likelihood $p(fe_j|fa_i)$ can be obtained, as shown in the following equation:

$$\tilde{p}(fe_j|fa_i) = n_{\text{det}}/n_{\text{all}} \quad (8)$$

where n_{all} denotes the total number of simulations with system fault fa_i and n_{det} denotes the number of simulations where feature $p(fe_j)$ is detected. This approximation value is used as the value of likelihood $p(fe_j|fa_i)$. Other likelihoods are obtained in the same way.

According to the law of total probability, the marginal probability $p(fe_j)$ is obtained as follows:

$$\begin{aligned} p(fe_j) &= p(fe_j|fa_i)p(fa_i) + p(fe_j|u_k)p(u_k) \\ &\quad + p(fe_j|(fa_i, u_k))p(fa_i, u_k) + p(fe_j|nor)p(nor) \end{aligned} \quad (9)$$

D. Proposed Probabilistic Fault Detection Function

In Fig. 4, an integrated fault detection function is proposed, integrating the conventional fault detection function, the proposed

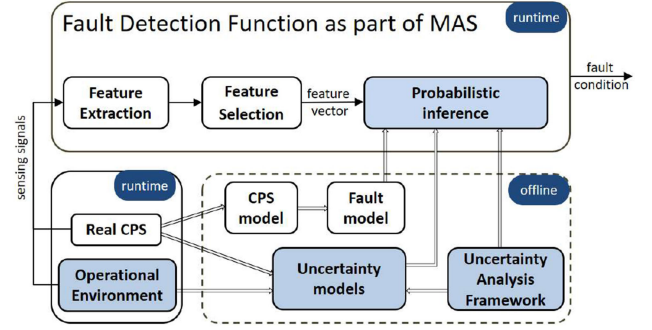


Fig. 4. Proposed fault detection function considering uncertainties.

uncertainty analysis framework, and the BI-based probability inference approach.

For a specific CPS in a certain operational environment, the CPS model, fault model, and uncertainty models are derived offline. An uncertainty analysis framework is developed at an abstract level for identifying the uncertainties in fault detection functions. Uncertainty models are built with information from both the system and the operational environment, as well as the uncertainty analysis framework. During runtime, information from the fault model, uncertainty model, and uncertainty analysis framework are integrated in the probabilistic inference.

Assuming that the feature extraction and selection algorithms are specified, the fault detection function can be presented as an algorithm, as shown in Algorithm 2. This algorithm summarizes the whole process of how the proposed fault detection function is realized.

E. Mathematical Performance Analysis of the Proposed Fault Detection Function

As illustrated in Section III-C, on condition that feature fe_j is detected, the probabilities of the four causes of the detection of this feature can be calculated using BI in (5). Since these four causes cover all the possibilities, the sum of the posterior probabilities satisfies the following equation:

$$p(fa_i|fe_j) + p(u_k|fe_j) + p((fa_i, u_k)|fe_j) + p(nor|fe_j) = 1 \quad (10)$$

If a fault detection function does not consider uncertainties, referred to as “conventional function” in this article, the output of the fault detection function is $p(fa_i|fe_j) = 1$. As a result, feature detections caused by uncertainties are also regarded as “system faults.”

In the proposed fault detection function, uncertainties both from the environment and the system are analyzed and modeled. Therefore, false detections of system faults caused by uncertainties are possible to be detected with prior knowledge or runtime monitoring of uncertainties. Having obtained the posterior probabilities of all the four possible causes, Algorithm 1 is further used for deciding the output of the function as “system fault” or “normal.” In this case, $p(fa_i|fe_j) < 1$, indicating the increase of fault detection accuracy.

Algorithm 2: Overall Fault Detection Algorithm.

```

01: Procedure Fault Detection
02: Initialize  $Fa, U, Fe$ ;
03: define  $p(fa_i)$  by empirical knowledge;
04: define  $p(u_k)$  by external monitors or empirical
    knowledge;
05: for every  $fa_i$  and  $u_k, i, j, k \in \mathbb{N}, i \leq I, j \leq J, k \leq K$ 
06:  $p(fa_i, u_k) \leftarrow p(fa_i)p(u_k)$ ;
07:  $p(nor) \leftarrow 1 - \left( \sum_{i \in I} p(fa_i) + \sum_{k \in K} p(u_k) \right. \right.$ 
     $\left. \left. + \sum_{i \in I, k \in K} p(fa_i, u_k) \right)$ ;
08: inject system fault and/or environment uncertainty
    into the Simulink model;
09: run simulation model;
10: run feature extraction algorithm;
11: if no feature detected
12: then output ‘normal’;
13: else calculate
     $p(fe_j|fa_i), p(fe_j|u_k), p(fe_j|(fa_i, u_k))$ ;
14: end for
15: calculate  $p(fe_j)$ 
16: calculate
     $p(fa_i|fe_j), p(u_k|fe_j), p((fa_i, u_k)|fe_j), p(nor|fe_j)$ 
17: for all  $j, k \in \mathbb{N}, j \leq J, k \leq K, \forall i \in \mathbb{N}, i \leq I$ 
18: if  $p(fa_i|fe_j) > p(u_k|fe_j)$  or
     $p((fa_i, u_k)|fe_j) > p(u_k|fe_j)$ 
19: and
     $p(nor|fe_j) < \min(p(fa_i|fe_j), p(u_k|fe_j), p((fa_i, u_k)|fe_j))$ 
20: then  $p(fa_i|fe_j) = 1$ 
21: else  $p(fa|fe_j) = 0$ 
22: end for
23: for all  $j \leq J$ 
24: if  $\forall_j p(fa|fe_j) = 1$ 
25: then output ‘system fault’
26: else output ‘normal’
27: end for
28: end procedure

```

IV. CASE STUDY AND DISCUSSION

In this section, first, the targeted CPS and the aim of the case study are clarified; second, the system fault and uncertainty are specified; and third, the proposed fault detection function is tested and results are presented and discussed.

A. Targeted CPS and the Aim of the Case Study

In advanced CPS, an MAS relies on sensing devices for environment perception and information gathering. In this article, a case study is conducted based on an MEMS accelerometer, which represents a CPS component.

This case study refers to our previous work in [38]. In [38], a Simulink model of an accelerometer was built as a

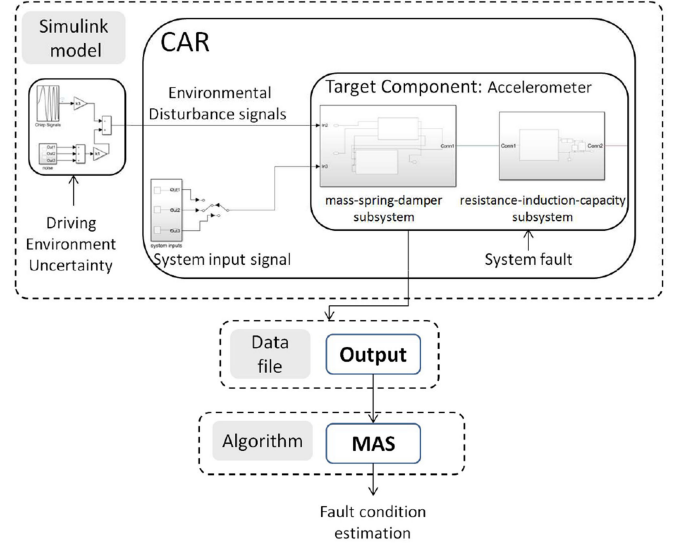


Fig. 5. Scenario description of the case study.

linear model containing a mechanical part based on a mass–spring–damper subsystem and an electrical part based on a resistance–induction–capacity subsystem. The model has force signal input and voltage signal output. The validity of the model was tested and proved satisfied. Several faults were injected into the model and the corresponding signal features of the output were analyzed and summarized in detail. In this case study, the accelerometer model and one of the explored faults in [38] were reused, as well as the corresponding signal feature. However, this case study involves more complex operational scenario with environment uncertainty, as shown in Fig. 5.

The operational scenario, built with Simulink model, is that an accelerometer is mounted in the electronic stability control system [39] of a car, and the car is accelerating. The input signal of the accelerometer is the force caused by the acceleration of the car, presented by a ramp signal. The validation of the proposed function, corresponding to the measurements, metrics, and questions in Table I, has three aspects as follows.

1) *Authenticity of the Problem Statement:* A fundamental assumption of this article is that the detected feature is potentially caused by uncertainties rather than system faults, which is validated in Section IV-C.

2) *Usability of the Fault Detection Function:* To ensure the usability of the proposed function, the feasibility of the proposed uncertainty quantification and parameter acquisition methods is validated in Section IV-D1.

3) *Performance of the Fault Detection Function:* Based on the accelerometer model in Section IV-A and the acquired parameters in Section IV-D1, the performance of the proposed function is validated in Section IV-D2.

B. Specification of Faults and Environment Uncertainty

1) *Fault and Feature:* The system fault injected into the accelerometer is represented by an extreme increased value of the inductance indicating the breakdown of the inductor. The

fault of the breakdown of inductor is chosen since it will cause a severe shift of the accelerometer output, which potentially causes risks for the system functionality. The fault injection has been conducted in a simulation environment with a Simulink model in our previous work in [38] and the corresponding signal feature is multiple dominant frequencies.

The feature used in the framework is chosen as the existence of multiple dominant frequencies in the output signal of the accelerometer model. Frequency analysis, including FFT, is a widely used and well-established method for fault detection. Related mathematical introduction can be found in [14]. To detect the feature, the frequency components of the output signal are first obtained by FFT, which are then filtered with thresholds. Then, the dominant frequency number is calculated. If the number is more than one, the multiple dominant frequencies exist, indicating that the feature is detected.

2) *Uncertainty*: The environment uncertainty, as a kind of uncertainty discussed in Section III-B, is presented by the existence of bumpy road condition. The force arising from such disturbance is presented by chirp signals. Chirp signals have complex and changing frequencies and are also used as the excitation of a bad country road in [40]. In a linear chirp signal, the instantaneous frequency varies linearly with time

$$f(t) = ct + f_0 \quad (11)$$

where f_0 is the starting frequency (at time $t = 0$) and c is the chirpiness.

In this case study, in the simulation environment, chirp signals, added with some low-level noise signals, are chosen to represent the environment uncertainty.

C. Validation of the Authenticity of the Problem Statement

Problem: Both system faults and uncertainties potentially result in the detection of a feature in the signals, which further causes false fault detection.

To represent the four possible causes of feature detection introduced in Section III-C, four cases of system condition are established as follows:

- Case 1*: system fault, no environment uncertainty;
- Case 2*: no system fault, environment uncertainty;
- Case 3*: system fault, environment uncertainty;
- Case 4*: no system fault, no environment uncertainty.

The force input and voltage output of the accelerometer model are shown in Fig. 6. In this figure, periodical components in the output signals appear in Cases 1–3, whereas the outputs and the inputs are linearly correlated in Case 4. The frequency distribution of the output signals obtained by the FFT is shown in Fig. 7, with multiple dominant frequencies in Cases 1–3. Since a feature of the output signals, multiple dominant frequencies, is detected in all three cases, the existence of system fault cannot be determined by such feature detection.

D. Validation of the Proposed Fault Detection Function

In this section, the proposed function is validated by providing probabilistic inference on system fault condition.

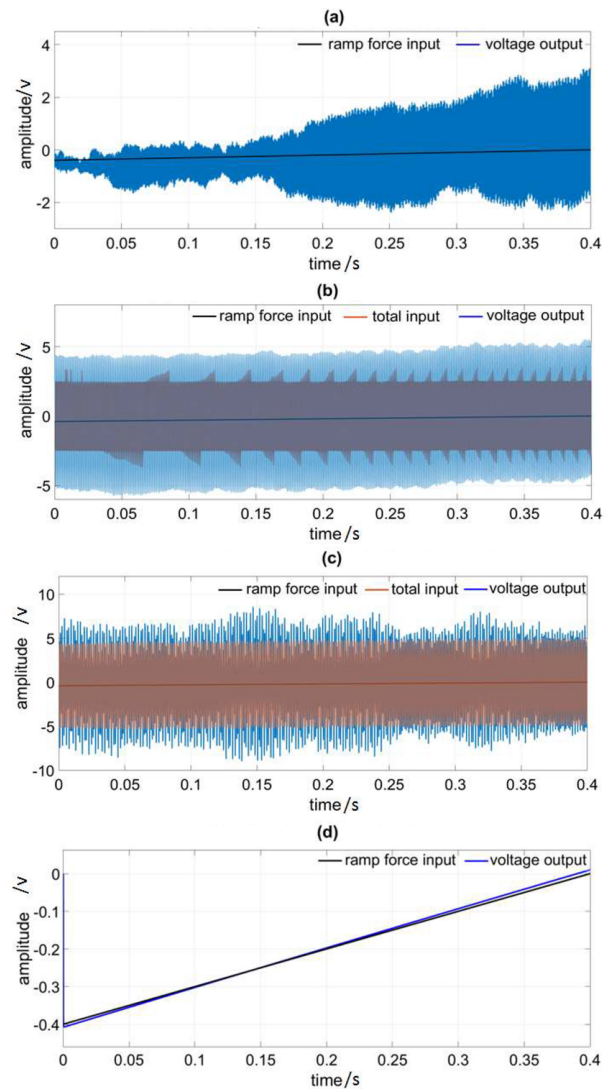


Fig. 6. Voltage output of the accelerometer model. (a) System input and output under Case 1. (b) System input and output under Case 2. (c) System input and output under Case 3. (d) System input and output under Case 4.

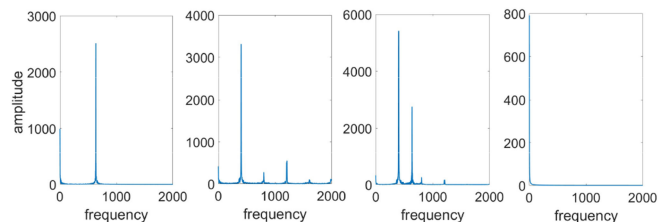


Fig. 7. Frequency distribution of the voltage output signals under Cases 1–4 from left to right.

1) *Probabilistic Parameter Acquisition*: The parameters of the fault detection function are acquired in various ways. As discussed in Section III-B, the prior probabilities are obtained by empirical knowledge or by other monitoring services. In this case study, the failure rate of the components is obtained by looking up related handbook. An example is given in [36]

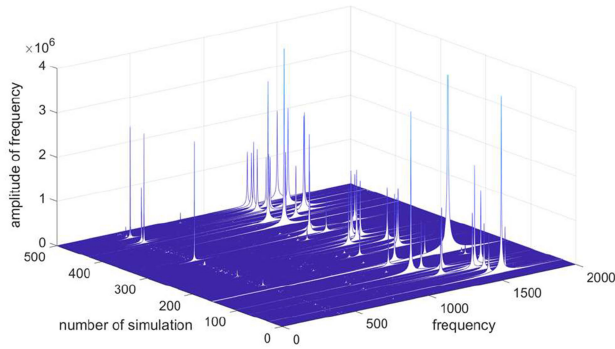


Fig. 8. Frequency distribution of the voltage output of the accelerometer model with system fault of inductor breakdown.

TABLE II
LIKELIHOOD VALUES OBTAINED BY SIMULATION

Item \ case	1	2	3	4
Total simulation number	500	500	500	500
Feature detected	446	160	238	0
Likelihood expression	$p(fe fa)$	$p(fe u)$	$p(fe (fa,u))$	$p(fe nor)$
Likelihood Value	0.89	0.36	0.47	0

that typical failure rate λ is 10^{-10} to $10^{-7}h^{-1}$ for electronic components at 40 °C, doubling for a temperature increase of 10 °C–20 °C. Considering an electronic system of a car, once the temperature profile around electronic control unit (ECU) X is confirmed, its failure rate can be obtained. Meanwhile, some marginal probabilities can be obtained with other MAS in runtime. For example, the probability that the road is bumpy can be assessed dynamically according to the real-time road condition.

The likelihoods $p(fe|fa)$, $p(fe|u)$, $p(fe|(fa,u))$, and $p(fe|nor)$ are obtained by simulations. In order to obtain the value of $p(fe|fa)$, 500 pairs of system parameters with high inductance value, representing the system fault, are assigned. Corresponding simulations with the accelerometer model are executed and the feature extraction algorithm is utilized to detect the features with the output voltage signals.

With the FFT, the frequency distribution of the output signals can be obtained, as shown in Fig. 8. Among the 500 simulations, many of them have multiple dominant frequencies in the output signals, shown as white peaks in the figure. For some simulations, the peaks are invisible in the figure because their amplitudes are relatively small. However, it is likely that there are also multiple dominant frequencies in the output signals of these simulations. So in our case, the number of dominant frequencies, instead of the amplitude of them, is chosen as the feature indicating system fault. With some postprocessing and thresholding methods, the numbers of dominant frequencies are obtained. If the number is more than 1, the value of the feature is 1; otherwise the value of the feature is 0. Using simulations, the likelihood values in different cases are obtained and listed in Table II. In this table, all the likelihoods are less than 1 in Cases

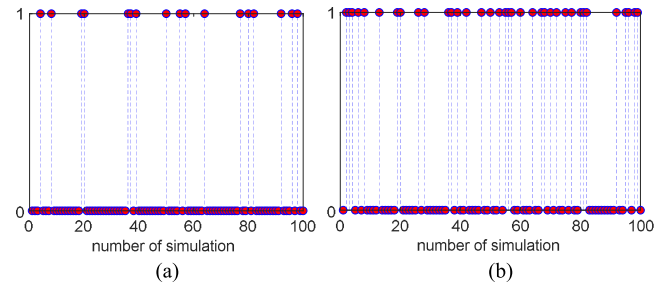


Fig. 9. Fault detection result with the proposed function and conventional function under Case 2. In the y-axis, “1” and “0” denote the detection result of “system fault” and “no system fault,” respectively. (a) Detection result with the proposed function. (b) Detection result with the conventional function.

1–3, which means that neither system fault nor environment uncertainty can guarantee feature detection.

2) *Fault Detection and Result Discussion*: In this section, the performance of the proposed fault detection function is explored. Corresponding to the four cases in Section IV-C, four groups of parameter values are assigned. The first group contains 100 simulations with system faults; the second group contains 100 simulations with environment uncertainty; the third group contains 100 simulations with both of them; and the last group contains 100 simulations with none of them. By executing the simulation models, the voltage outputs are obtained and input into the fault detection function. As the output of the function, a fault condition noted as “system fault” or “normal” is obtained.

As a comparison, a conventional fault detection function is also used. In this function, a feature extraction algorithm is first applied to the voltage signal outputs. This algorithm is the same as the feature extraction algorithm used in the proposed function and is described with details in Section IV-B1. After the feature detection process, the feature-fault mapping is conducted. If and only if the feature is detected, fault condition “system fault” is exported. Otherwise, fault condition “normal” is exported. The detection result in Case 2 is shown in Fig. 9. In this case, there is no system fault while the environment uncertainty is high. The detection results are expected to be “normal,” denoted as 0 in the y-axis of the figure. From the figure, it is shown that both the conventional function and the proposed function have false detection cases, presented by the red dots with the y-axis value of 1. It is clearly shown that the proposed function has much less false detection than the conventional function. The consideration of uncertainties of environment uncertainty contributes to the decrease of false detection.

The detection accuracy, calculated as the ratio of correct detections and total detections, is shown in Fig. 10. In Case 1, the detection accuracies of both functions are 89%, which is relatively high compared with Cases 2 and 3. In this case, the false detections mainly result from the system itself, including the uncertainty of feature extraction algorithm and fault model as discussed in Section III-B. In Case 2, the detection accuracy increases significantly from 64% to 83% with the application of the proposed function. In this case, there are no system faults while the environment uncertainty is severe, so parts of the false detections are avoided by the proposed function. In Case 3,

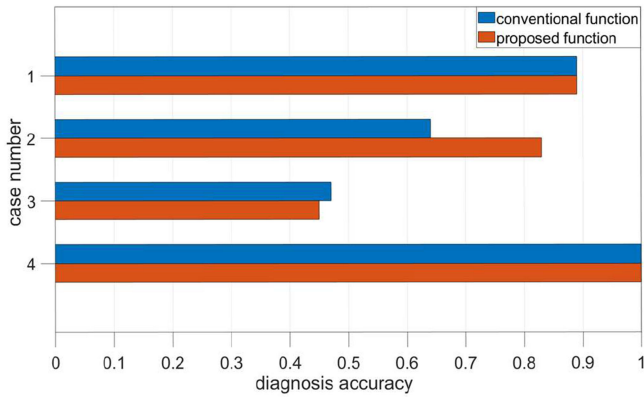


Fig. 10. Detection accuracy with the proposed function and conventional function under four cases.

where there are system faults and environment uncertainty, the detection results are expected to be “system fault.” In this case, the accuracies of the two functions are both around 45%, which is quite low, since it is difficult to distinguish the root cause of feature detection under this condition. As shown in the figure, the detection accuracy of conventional function is even slightly higher than that of the proposed function. One possible reason for this phenomenon is that the proposed function is more likely to classify those scenarios with high environment uncertainty as “normal.” In Case 4, where there is no environment uncertainty and system fault, the detection accuracies of both functions are 100%. Overall, the proposed function has improved the detection accuracy of system fault under uncertainty of severe environment uncertainty.

V. CONCLUSION AND FUTURE WORK

In this article, we have addressed the challenges of fault detection in CPS in the presence of uncertainties. The fault detection function proposed in this article provides probabilistic inference of system faults based on BI considering heterogeneous uncertainties. An uncertainty analysis framework specialized for the fault detection process has significantly concretized the meaning of uncertainties, the way to quantify them, and the way to propagate them into the function. The probabilistic parameter values of the function are acquired through simulation and empirical knowledge. The functionality of the function has been validated using a case study of an accelerometer model, and the fault detection accuracy is evidently improved. When this function is implemented in MAS services, an online fault condition monitoring is achieved with various data inputs from other subsystems of CPS.

However, due to the complex nature of CPS and various operational environments, the proposed uncertainty analysis framework and fault detection function have certain delimitations and more efforts are needed for improvement in the future. For example, an assumption is made in this article that system faults and uncertainties are independent, which is not valid in some complex situations. Meanwhile, the validation process only involves one system fault and one feature in the case

study, which is designed for validating the usability of the proposed fault detection function. In the future, the way faults and uncertainties propagate, and the compositional effect of various types of system faults and uncertainties from different subsystems will be explored. Moreover, experiments and validation with more complex CPS applications other than accelerometer will be conducted.

REFERENCES

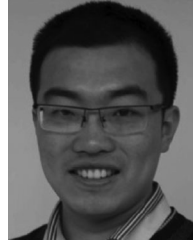
- [1] E. R. Griffor, C. Greer, D. A. Wollman, and M. J. Burns, “Framework for cyber-physical systems: Volume 1, overview,” *Nat. Inst. Standards Technol.*, Gaithersburg, MD, USA, Rep. 1500-201, Jun. 2017.
- [2] S. K. Khaitan and J. D. McCalley, “Design techniques and applications of cyberphysical systems: A survey,” *IEEE Syst. J.*, vol. 9, no. 2, pp. 350–365, Jun. 2015.
- [3] A. Banerjee, K. K. Venkatasubramanian, T. Mukherjee, and S. K. S. Gupta, “Ensuring safety, security, and sustainability of mission-critical cyber-physical systems,” *Proc. IEEE*, vol. 100, no. 1, pp. 283–299, Jan. 2012.
- [4] D. Chen and Z. Lu, “A model-based approach to dynamic self-assessment for automated performance and safety awareness of cyber-physical systems,” in *Proc. Int. Symp. Model-Based Saf. Assessment*, 2017, pp. 227–240.
- [5] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, “Basic concepts and taxonomy of dependable and secure computing,” *IEEE Trans. Depend. Secure Comput.*, vol. 1, no. 1, pp. 11–33, Jan. 2004.
- [6] P. E. Lanigan, S. Kavulya, P. Narasimhan, T. E. Fuhrman, and M. A. Salman, “Diagnosis in automotive systems: A survey,” Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-PDL-11-110, 2011.
- [7] H. Jiang, J. J. Zhang, W. Gao, and Z. Wu, “Fault detection, identification, and location in smart grid based on data-driven computational methods,” *IEEE Trans. Smart Grid*, vol. 5, no. 6, pp. 2947–2956, Nov. 2014.
- [8] S. Ofsthun, “Integrated vehicle health management for aerospace platforms,” *IEEE Instrum. Meas. Mag.*, vol. 5, no. 3, pp. 21–24, Sep. 2002.
- [9] J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*. Evanston, IL, USA: Routledge, 2017.
- [10] L. Xing and S. V. Amari, “Fault tree analysis,” in *Handbook of Performance Engineering*. London, U.K.: Springer, 2008, pp. 595–620.
- [11] J. W. Sheppard and S. G. W. Butcher, “A formal analysis of fault diagnosis with D-matrices,” *J. Electron. Testing*, vol. 23, no. 4, pp. 309–322, Aug. 2007.
- [12] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, Jul. 2009.
- [13] R. Chalapathy and S. Chawla, “Deep learning for anomaly detection: A survey,” 2019, *arXiv:1901.03407*.
- [14] M. Gerdes, D. Galar, and D. Scholz, “Genetic algorithms and decision trees for condition monitoring and prognosis of A320 aircraft air conditioning,” *Insight—Non-Destruct. Testing Condition Monit.*, vol. 59, no. 8, pp. 424–433, Aug. 2017.
- [15] F. Ye, Z. Zhang, K. Chakrabarty, and X. Gu, “Board-level functional fault diagnosis using artificial neural networks, support-vector machines, and weighted-majority voting,” *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 32, no. 5, pp. 723–736, May 2013.
- [16] P. S. Pillai and S. Rao, “Resource allocation in cloud computing using the uncertainty principle of game theory,” *IEEE Syst. J.*, vol. 10, no. 2, pp. 637–648, Jun. 2016.
- [17] M. Zhang, B. Selic, S. Ali, T. Yue, O. Okariz, and R. Norgren, “Understanding uncertainty in cyber-physical systems: A conceptual model,” in *Proc. Eur. Conf. Model. Found. Appl.*, 2016, vol. 1, pp. 247–264.
- [18] M. Törngren and U. Sellgren, “Complexity challenges in development of cyber-physical systems,” in *Principles of Modeling* (Lecture Notes in Computer Science). New York, NY, USA: Springer, 2018, pp. 478–503.
- [19] B. Taylor and C. Kuyatt, “Guidelines for evaluating the uncertainty of NIST measurement results,” Nat. Inst. Standards Technol., Gaithersburg, MD, USA, NIST Tech. Note 1297, 1994.
- [20] A. Der Kiureghian and O. Ditlevsen, “Aleatory or epistemic? Does it matter?” *Struct. Saf.*, vol. 31, no. 2, pp. 105–112, Mar. 2009.
- [21] T. Aven, P. Baraldi, R. Flage, and E. Zio, *Uncertainty in Risk Assessment: The Representation and Treatment of Uncertainties by Probabilistic and Non-Probabilistic Methods*. Hoboken, NJ, USA: Wiley, 2013.
- [22] P. Baraldi, M. Compare, and E. Zio, “Uncertainty treatment in expert information systems for maintenance policy assessment,” *Appl. Soft Comput.*, vol. 22, pp. 297–310, Sep. 2014.

- [23] B. Cai, L. Huang, and M. Xie, "Bayesian networks in fault diagnosis," *IEEE Trans. Ind. Inform.*, vol. 13, no. 5, pp. 2227–2240, Oct. 2017.
- [24] Z. Zhang, Z. Wang, X. Gu, and K. Chakrabarty, "Board-level fault diagnosis using Bayesian inference," in *Proc. IEEE Very Large Scale Integr. Test Symp.*, 2010, pp. 244–249.
- [25] H. Habibi, I. Howard, and R. Habibi, "Bayesian sensor fault detection in a Markov jump system," *Asian J. Control*, vol. 19, no. 4, pp. 1465–1481, 2017.
- [26] N. Fenton and M. Neil, "The use of Bayes and causal modelling in decision making, uncertainty and risk," *CEPIS Upgrade*, vol. 12, no. 5, pp. 10–21, 2011.
- [27] S. Krishnamurthy, S. Sarkar, and A. Tewari, "Scalable anomaly detection and isolation in cyber-physical systems using Bayesian networks," in *Proc. ASME Dyn. Syst. Control Conf.*, San Antonio, TX, USA, 2014.
- [28] A. Ogbechie, J. Díaz-Rozo, P. Larrañaga, and C. Bielza, "Dynamic Bayesian network-based anomaly detection for in-process visual inspection of laser surface heat treatment," in *Machine Learning for Cyber Physical Systems*. Berlin, Germany: Springer, 2017, pp. 17–24.
- [29] J. Ying, T. Kirubarajan, K. R. Pattipati, and A. Patterson-Hine, "A hidden Markov model-based algorithm for fault diagnosis with partial and imperfect tests," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 30, no. 4, pp. 463–473, Nov. 2000.
- [30] I. Meedeniya, I. Moser, A. Aleti, and L. Grunske, "Evaluating probabilistic models with uncertain model parameters," *Softw. Syst. Model.*, vol. 13, no. 4, pp. 1395–1415, Oct. 2014.
- [31] R. D. Arnold and J. P. Wade, "A definition of systems thinking: A systems approach," *Proc. Comput. Sci.*, vol. 44, pp. 669–678, 2015.
- [32] B. Li, P. Zhang, H. Tian, S. Mi, D. Liu, and G. Ren, "A new feature extraction and selection scheme for hybrid fault diagnosis of gearbox," *Expert Syst. Appl.*, vol. 38, no. 8, pp. 10000–10009, Aug. 2011.
- [33] J. Li *et al.*, "Feature selection: A data perspective," *ACM Comput. Surv.*, vol. 50, no. 6, pp. 1–45, Dec. 2017.
- [34] M. Pechta, D. Dasa, and A. Ramakrishnan, "The IEEE standards on reliability program and reliability prediction methods for electronic equipment," *Microelectron. Rel.*, vol. 42, no. 9–11, pp. 1259–1266, Sep. 2002.
- [35] M. G. Pecht and F. R. Nash, "Predicting the reliability of electronic equipment," *Proc. IEEE*, vol. 82, no. 7, pp. 992–1004, Jul. 1994.
- [36] A. Birolini, "Reliability engineering," in *Lees' Loss Prevention in the Process Industries*. New York, NY, USA: Elsevier, 2012, pp. 131–203.
- [37] G. Grimmett and D. Stirzaker, *Probability and Random Processes*. London, U.K.: Oxford Univ. Press, 2001.
- [38] X. Tao, D. Chen, and J. Sagarduy, "Signal feature analysis for dynamic anomaly detection of components in embedded control systems," in *Int. Conf. Dependability Complex Syst.*, 2019, pp. 471–481.
- [39] H. Fennel and E. L. Ding, "A model-based failsafe system for the continental TEVES electronic-stability-program (ESP)," SAE Int., Warrendale, PA, USA, SAE Tech. Paper 2000-01-1635, 2000.
- [40] S. Spirk and K.-U. Henning, "Wheel load oriented control of semi-active and active suspension systems using pre-located road sampling," in *Proc. FISITA World Automotive Congr.*, 2013, vol. 201, pp. 167–182.



Xin Tao received the B.S. degree in mechanical engineering from the Wuhan University of Technology, Wuhan, China, in 2014, and the M.S. degree in digital signal processing from the University of Science and Technology of China, Hefei, China, in 2017. She is currently working toward the Ph.D. degree with the Division of Machine Design, KTH Royal Institute of Technology, Stockholm, Sweden.

Her research interests include system monitoring, fault detection and uncertainty management of cyber-physical systems.



Jinzhi Lu received the Ph.D. degree from the Mechatronics Division, KTH Royal Institute of Technology, Stockholm, Sweden, in 2019.

He is a Certified Systems Engineering Professional. He is currently a Research Scientist with the École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. His research interests include model-based systems engineering (MBSE) tool-chain design and MBSE enterprise transitioning.

Dr. Lu is Senior Member of the China Council on Systems Engineering.



Dejiu Chen (Senior Member, IEEE) received the M.Sc. degree in mechatronics and Ph.D. degree in research on embedded computer control system architecture from KTH Royal Institute of Technology, Stockholm, Sweden, in 1998 and 2004, respectively.

He is currently an Associate Professor in embedded control systems with the KTH Royal Institute of Technology, Stockholm, Sweden. He was with the Enea Data AB, Stockholm, Sweden, as a Senior Technical Instructor. His research interests include cyber-physical systems in the areas of model-based

engineering, architecture design and quality management, safety engineering, situation awareness, and self-management.



Martin Törngren (Senior Member, IEEE) received the M.Sc. degree in mechanical engineering and the Ph.D. degree in mechanical engineering with a specialization in mechatronics from the KTH Royal Institute of Technology, Stockholm, Sweden, in 1987 and 1995, respectively.

In 1995, he co-created the company Fengco Real Time Control. He became an Assistant Professor at KTH in 1996, did a postdoc at the EU-JRC Dependability Unit, Ispra, Italy, in 1998, and became a Docent in 1999 at KTH and Full Professor at KTH in embedded control systems in 2002. He is the Principal Initiator and Director of the KTH Innovative Centre for Embedded Systems launched in 2008. In 2011–2012, he was a Visiting Scholar with the Department of Electrical Engineering and Computer Sciences, UC Berkeley, and did a two-month sabbatical in the spring 2018 at Stevens Institute of Technology, Hoboken, NJ, USA. His research interests include cyber-physical systems, architectural design, system safety, model-based engineering, and codesign of control applications and embedded systems.

Prof. Törngren is the recipient of the ARTEMIS Recognition Award in 2013 to the iFEST Project (as a Project Technical Coordinator), the ITEA Achievement Award in 2004 for contributions in the Embedded Electronic Architecture for the European Automotive Industry (EAST-EAA) Project, the Swedish Aerospace and Defence Company (SAAB)-Scania Award in 1994 for qualified contributions in distributed control systems, and several best paper awards, the most recent one from International Symposium on Industrial Embedded Systems (SIES) 2017.