Fault Detection of Discrete-Time Stochastic Systems Subject to Temporal Logic Correctness Requirements

Jun Chen, *Member, IEEE*, and Ratnesh Kumar, *Fellow, IEEE*

***Abstract*—This paper studies the fault detection of discrete-time stochastic systems with linear-time temporal logic (LTL) as correctness requirement—A fault is a violation of LTL specification. The temporal logic allows system correctness properties to be specified compactly and in a user-friendly manner (being close to natural-languages), and supports automatic translation into other formal models such as automata. We introduce the notion of input-output stochastic hybrid automaton (I/O-SHA) and show that the refinement of a continuous physical system (modeled as stochastic difference equations) against a certain class of LTL correctness requirement can be modeled as an I/O-SHA. The refinement preserves the behaviors of the physical system and also captures requirement-violation as a reachability property. Probability distribution over the discrete locations of hybrid system is estimated recursively by computing the distributions for continuous variables for each discrete location. This is then used to compute the *likelihood of fault*, a statistic that we employ for the purpose of fault detection. The performance of the detection scheme is measured in terms of false alarm (FA) and missed detection (MD) rates, and the condition for the existence of a detector to achieve any desired rates of FA and MD is captured in form of *Stochastic-Diagnosability*, a notion that we introduce in this paper for stochastic hybrid systems. The proposed method of fault detection is illustrated by a practical example.**

***Note to Practitioners*—Many cyberphysical systems, such as building automation systems, automotive vehicles and smart power grids, can be modeled as stochastic systems with mixed continuous and discrete dynamics subject to disturbance and noise, whose behaviors are monitored and controlled by networked (digital) control systems. This paper investigates fault detection in the form of temporal logic specification violation in model-based approach, by transforming it into a state estimation problem for stochastic system models. We provide an algorithm for online fault detection and introduce the notion of Stochastic-Diagnosability for the existence of a detector with any desired accuracy of detection as measured by false alarms and missed detections. The work is illustrated by a room heating system example.**

***Index Terms*—Bayesian filtering, cyberphysical systems, diagnosability, fault detection, linear-time temporal logic, stochastic hybrid systems.**

Manuscript received January 07, 2014; revised March 24, 2015; accepted June 25, 2015. Date of publication July 27, 2015; date of current version October 02, 2015. This paper was recommended for publication by Associate Editor K. Akesson and Editor S. Reveliotis upon evaluation of the reviewers' comments. The work was supported in part by the National Science Foundation under the Frant NSF-ECCS-0801763, Grant NSF-ECCS-0926029, and Grant NSF-CCF-1331390. The work of J. Chen was performed while he was with Iowa State University.

J. Chen was with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA. . He is now with Idaho National Laboratory, Idaho Falls, ID 83415 USA (e-mail: junchen@iastate.edu).

R. Kumar is with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: rkumar@iastate.edu).

Digital Object Identifier 10.1109/TASE.2015.2453193

I. Introduction

DETECTING system failures is an important and challenging problem in many disciplines such as software engineering [1], automotive systems [2], power systems [3], nuclear engineering [4], aerospace engineering [5], and digital circuits [6]. In general, a fault is a deviation of a system from its required or normal behavior, such as executing a fault-event, reaching a fault-state, or violating a system specification, and needs to be detected accurately within a tolerable delay bound to ensure timely activation of any fault tolerance actions.

The problem of fault detection has been widely researched [7]–[18]. Reference [10] considers the fault detection in stochastic discrete event systems (DESs), for which the condition for the existence of a detector for achieving any desired accuracy requirement has been captured as *stochastic diagnosability*, as studied in [12]–[14]. For dynamical systems, a fault can be modeled as an *abrupt* or *steady* change in system dynamics through a change in system parameters (e.g., [7], [8]), or appearance of *additive* terms in the state equation (e.g., [15] and [16]). The fault diagnosis problem is also studied for hybrid systems [17], [18]. Reference [18] considers hybrid systems whose continuous dynamics are restricted to first-order, and a fault can abruptly change both continuous and discrete dynamics. The model-based diagnosis techniques and signature analysis are integrated in the proposed diagnosis algorithm of [18]. The problem of runtime verification of LTL formula for nonstochastic can be found in [19]–[21].

In this paper, we study fault detection of certain physical systems, whose dynamics over discrete sample instances are described by stochastic difference equations, and in contrast to the aforementioned works, we consider a more general notion of a fault, namely a violation of certain correctness requirements expressed as linear-time temporal logic (LTL) formulas. The results presented in this paper can be straightforwardly adopted to the case where a fault is defined to be a change in system dynamics (see Remark 7). LTL formulas are widely used as correctness requirements [22]–[25], as they are easier to specify than automata models or ω-language, yet they are compact and expressive and support automatic translation into automata/formal-language models. In [22], [23], the authors studied fault diagnosis problem in the setting of deterministic DESs whose nonfault behaviors are expressed as LTL formulas; the present work generalizes to stochastic discrete-time systems. References [24], [25] consider the controller design problem for hybrid/piecewise-affine systems, where the desired system behaviors are subject to LTL requirements.

An LTL formula is defined over infinite traces, so any system behavior violating a given formula will be of infinite length.

Since a fault detector can access only a finite history of observations, it is natural to assume that every infinite run of a system, that violates a given LTL formula, possesses a finite prefix, called an *indicator*, such that all its infinite extensions that are feasible in the system also violate the LTL formula. This is a necessary requirement for any online diagnosis, and a system is said to be *prediagnosable* if it satisfies such a property with respect to the given LTL specification [22]. As established in [22], prediagnosability is equivalent to the existence of a deterministic Büchi automaton which accepts those system traces satisfying the LTL property, and those equal the limits of finite prefixes accepted by the same model. To aid the diagnosis analysis, we refine the continuous physical system against the Büchi automaton to obtain a stochastic hybrid system for which fault detection problem becomes a stochastic reachability estimation problem.

Stochastic hybrid systems are widely studied in literature [26]–[32]. For example, continuous time stochastic hybrid system is studied in [30], whereas probabilistic reachability problem for discrete time stochastic hybrid system is considered in [26], [29]. The abstraction of a stochastic hybrid system is examined in [27], [28], with a goal to find a relaxed model which is computationally simpler and possesses behaviors with bounded deviations compared to the original concrete system. Model checking of temporal properties for stochastic hybrid systems has been examined in [31], [32].

To model the refinement of the continuous physical system against the Büchi requirement model, we introduce the notion of input–output stochastic hybrid automaton (I/O-SHA), extending the logical input–output hybrid automaton (I/O-HA) introduced in [33], by allowing randomness in invariants, guards, data updates, and output assignments. Next we propose an algorithm that performs the refinement to yield an I/O-SHA such that the violation of the LTL formula is captured as a reachability property to a certain fault-location. The likelihood of fault versus no-fault (requirement-violation versus non-violation) is recursively computed and is used as a statistic for issuing fault detection decisions: Whenever the likelihood of fault arises above a suitable threshold, i.e., the likelihood of fault is "high", a fault decision is issued, and otherwise the detector remains silent. The performance of this detection scheme is determined by introducing and computing its false alarm (FA) and missed detection (MD) rates. In order to identify the class of systems for which detection with any desired accuracy is feasible, we introduce the notion of *Stochastic-Diagnosability* as the corresponding necessary and sufficient condition. The proposed diagnosis framework is implemented for a benchmark room heating problem, inspired from [34], to demonstrate the validity and applicability of the results.

The contributions of this paper are summarized as follows.

1) The notion of I/O-SHA is introduced, extending its logical counterpart in [33].
2) Stochastic filtering equations are provided to recursively update the distributions for both continuous states and discrete locations of I/O-SHA.
3) Fault detection, namely, LTL requirement-violation detection, is performed based on above stochastic filtering equations.
4) To measure the performance of our detection scheme we introduce the notions of false alarm and missed detection rates, and show their dependence on detection threshold and detection delay.
5) The notion of Stochastic-Diagnosability is introduced, and its necessity and sufficiency for the existence of a detector for achieving any desired false alarm and missed detection rates is established.

The remainder of this paper is organized as follows. Section II provides some preliminaries on LTL, while Section III formulates the fault detection problem. Definition of I/O-SHA is introduced in Section IV, where the main results are also given. An illustration by applying our results to a room heating problem is studied in Section V. Section VI introduces the notion of Stochastic-Diagnosability as an existence condition of a detector. The paper is concluded in Section VII.

II. Preliminaries

In this paper, we study fault detection of physical systems subject to disturbance and noise, modeled by stochastic difference equations

$$x_{k+1} = f(x_k, u_k, v_k) \tag{1}$$

$$r_k = g(x_k, u_k) \tag{2}$$

$$y_k = h(x_k, u_k, w_k) \tag{3}$$

where u, x, r, y, v, w represent, respectively, the input, state, requirement (unobserved), output (observed), disturbance and noise variables, and k is the time-index. Note that the requirement variable is used to capture a user-defined specification that, at each step, depends on system state and input, and, being a user-defined requirement, it is independent of disturbance or noise. The properties of the nonfault system behaviors are described by using a LTL formula over the requirement variables, which may not be directly observed and hence must be estimated from observations of inputs and outputs. In the following we present a brief description of LTL; a more thorough introduction can be found in [35], [36].

Let $M_d = (L_d, \delta, AP, label)$ be a state transition graph, where L_d is the set of states, $\delta : L_d \to 2^{L_d}$ is a total transition relation, i.e., $\forall l \in L_d, \delta(l) \neq \emptyset$, AP is a finite set of atomic proposition symbols, and $label : L_d \to 2^{AP}$ is a function that labels each state with the set of atomic propositions true at that state. A sequence of states $\pi = l_0(\pi)l_1(\pi)\ldots$ is a *state-trace* in M_d if $l_{i+1}(\pi) \in \delta(l_i(\pi))$ for every $i \in \{0, 1, \ldots\}$. $\pi^k = l_k(\pi)l_{k+1}(\pi)\ldots$, where $k \in \mathbb{N}$, is used to denote the suffix of π starting from index k. A *proposition-trace* over an atomic proposition set AP is defined as a sequence of set of atomic propositions, $\pi_p = label_0 label_1 \ldots$ such that $label_i \subseteq AP, \forall i \in \{0, 1, \ldots\}$. A proposition-trace $\pi_p = label_0 label_1 \ldots$ over AP is said to be *contained* in M_d if there exists a state-trace $\pi = l_0 l_1 \ldots$ in M_d such that $label_i = label(l_i), \forall i \in \{0, 1, \ldots\}$, in which case π_p is said to be associated with π.

LTL temporal logic is a formalism for describing properties of sequences of states. Such properties are expressed using *temporal operators* of the temporal logic which include:

- X ("next time"): it requires that a property holds in the next state of the state-trace;

- U ("until"): it is used to combine two properties. The combined property holds if there is a state on the state-trace where the second property holds, and at every preceding state on the trace, the first property holds;
- F ("eventually" or "in the future"): it requires that a property will hold at some future state on the state-trace;
- G ("always" or "globally"): it requires that a property holds at every state on the trace;
- B ("before"): it combines two properties. It requires that if there is a state on the state-trace where the second property holds, then there exists a preceding state on the trace where the first property holds.

We have the following relations among the above operators, where ϕ denotes a temporal logic formula.

- $F\phi \equiv trueU\phi$.
- $G\phi \equiv \neg F\neg\phi$.
- $\phi Bg \equiv \neg(\neg\phi Ug)$.

Thus, we can use X and U to express all of the other temporal operators. LTL formulas are generated by the following rules.

P1) If $p \in AP$, then p is a LTL formula.

P2) If ϕ_1 and ϕ_2 are LTL formulas, then so are $\neg\phi_1$ and $\phi_1 \wedge \phi_2$.

P3) If ϕ_1 and ϕ_2 are LTL formulas, then so are $X\phi_1$ and $\phi_1 U\phi_2$.

The semantics of LTL can be defined with respect to *infinite* state-traces in a state transition graph $M_d = (L_d, \delta, AP, label)$. For a LTL formula ϕ, we use the notation $< M_d, \pi > \models f$ (resp., $< M_d, \pi > \not\models f$) to denote that f holds (resp., does not hold) along the infinite state-trace π in M_d. The relation $\models$ is defined inductively as follows.

1) $\forall p \in AP, \pi \models p$ if and only if $p \in label(l_0(\pi))$.
2) $\pi \models \neg\phi$ if and only if $\pi \not\models \phi$.
3) $\pi \models \phi_1 \wedge \phi_2$ if and only if $\pi \models \phi_1$ and $\pi \models \phi_2$.
4) $\pi \models X\phi$ if and only if $\pi^1 \models \phi$.
5) $\pi \models \phi_1 U\phi_2$ if and only if there exists a k such that $\pi^k \models \phi_2$ and for all $j \leq k-1, \pi^j \models \phi_1$.

The semantics of LTL formulas can also be expressed over infinite length proposition-traces without referring to any specific state transition graph. This is done by replacing the first condition shown previously with

$$\forall p \in AP, \pi_p = label_0 label_1 \ldots \models p \Leftrightarrow p \in label_0$$

where π_p is an infinite proposition-trace over AP, i.e., $label_i \subseteq AP$ for all $i \geq 0$.

Given an LTL formula ϕ, denote S_ϕ as the set of all infinitely long proposition-traces over AP satisfying ϕ. Then, we can obtain a generalized nondeterministic Büchi automaton T_ϕ ([35]) that accepts S_ϕ. To construct T_ϕ, we first put ϕ into *negation normal form*, in which negation is only applied at atomic level. Then we rewrite each subformula of the form Fg, Gg, or $g_1 Bg_2$ as $TrueUg$. Let $|\phi|$ be the number of subformulas of the form $\lambda U\mu$. Then, the generalized nondeterministic Büchi automaton has $|\phi|$ sets of accepting states and is of the form

$$T_\phi = (L_\phi, 2^{AP}, \delta_\phi, l_0^\phi, \mathcal{L}_\phi)$$

where

- L_ϕ is the set of states;
- $\delta_\phi : L_\phi \times 2^{AP} \to L_\phi$ is the transition relation;
- l_0^ϕ is the initial state;
- $\mathcal{L}_\phi \subseteq 2^{L_\phi}$ is the generalized Büchi acceptance condition, such that for each subformula of the form $\lambda U\mu$ in ϕ, there exists a $\mathcal{L} \in \mathcal{L}_\phi$ which is used to capture the fulfillment of $\lambda U\mu$.

When $|\mathcal{L}_\phi| = 1$, then the generalized Büchi automaton reduces to a standard one. An infinite length proposition-trace $\pi_p = label_1 label_2 \ldots$ over AP is accepted by T_ϕ if and only if there exists an infinite length state-trace $\pi = l_0^\phi l_1 \ldots$ in T_ϕ such that $l_i \in \delta_f(l_{i-1}, label_i)$ for all $i \geq 1$, and π visits each set of locations in $\mathcal{L}_\phi$ infinitely often. Then the set of all infinite length proposition-traces accepted by T_ϕ, called its ω-*language*, equals S_ϕ.

While every LTL formula can be characterized as the ω-language accepted by a nondeterministic Büchi automaton, only certain fragments of LTL can be modeled as the ω-language accepted by a *deterministic* Büchi automaton. Since a detector has access to only a finite history of observed behavior, only the failure behaviors possessing finite *indicator* (see Definition 1) can be detected. It turns out that in this case, the accepted traces of the system satisfying the LTL specification can be modeled as the ω-language accepted by a deterministic Büchi automaton.

III. Fault Detection Problem Formulation

Suppose the dynamics of the physical system G under diagnosis can be described by the stochastic difference (1)–(3), where we recall that u, x, r, y, v, w represent, respectively, the input, state, requirement (unobserved), output (observed), disturbance, and noise variables, and k is the time-index. The initial state x_0, the *disturbance* v_k as well as the *noise* w_k are all assumed mutually independent with known distributions, which can be dependent on current states. Note that the requirement variable, which specifies a required value for each input-state pair through the function g, is used to capture a user-defined specification that, at each step, depends on system state and input, and being a user-defined requirement, it is not corrupted by noise. We assume that the properties of the required system behaviors can be described by using a LTL formula ϕ involving *predicates* defined over the requirement variables r_k, $k \in N$. Then the predicates, appearing in the LTL formula, and their Boolean combinations act as atomic propositions guarding the transitions in the Büchi automaton. The set of all infinitely long feasible sequences of aforementioned predicates is denoted as A_G.

Since detection of requirement-violation must occur based on a finite history of input/output observations, it is natural to assume that every infinite run of a system, that violates the given LTL formula, possesses a finite prefix, called an *indicator*, such that all of its infinite extensions that are feasible in the system also violate the LTL formula. This property was captured under the name of *prediagnosability* in [22], and is a *necessary* condition for any detector's ability to detect the violation of the specified LTL formula based on finite-length observations. So, without loss of generality, we assume that the prediagnosability holds. Next, we provide a formal definition of indicator and also of prediagnosability.

Definition 1: Given a system G and a LTL formula ϕ, a finite sequence of requirement variables is said to be an *indicator* if all of its infinite extensions in G violate ϕ. We denote the set of all indicators as $I_\phi(G)$. G is said to be *prediagnosable* with respect to ϕ if each infinite sequence of requirement variables violating ϕ possesses a finite prefix that is an indicator.

Remark 1: By utilizing the notion of indicator, detecting the occurrence of infinite trace violating a LTL formula is transformed into detecting the execution of finite indicators. As mentioned in [22], when an indicator is executed, the actual fault may not have happened yet. Hence, our framework includes both cases of fault detection (that a fault has already occurred) and prediction (that a fault will inevitably occur). Note that the notion of indicator has also been utilized for the purpose of fault prognosis (see for example [37]–[39]), where the prediction of a future fault is performed by detecting the occurrence of a non-fault prefix indicator. ■

Remark 2: Note that a system is inherently prediagnosable if the LTL formula ϕ is a safety one [35], i.e., it only requires that some "bad" things must never occur. However, when the correctness requirement is a more general one, the system may not be prediagnosable (See Example 1), and in this case, the violation of ϕ can not be detected even if the system is perfectly observable, i.e., $y_k = r_k$ for all $k \in \mathbb{N}$. For this reason, we assume without loss of generality that the system is prediagnosable with respect to the LTL formula. ■

As established in [22, Theorem 1], the prediagnosability of system G with respect to a LTL formula ϕ is equivalent to the existence of a deterministic Büchi automaton accepting $S_\phi \cap A_G$, which can also be characterized as the *limits* of the finite prefixes accepted by the same model treated as a standard finite state automaton. Then owing to the determinism, and assuming no redundant states, we can augment the Büchi automaton, by adding an absorbing state "F", reaching which indicates the execution of either a fault that already occurred or a future fault that is inevitable, thereby yielding an augmented deterministic requirement model, denoted as R. Note the augmentation requires adding one new transition from each state to the newly added fault state F, guarded by the complement of the existing transition guards of the state.

Example 1: Consider a system G with dynamics

$$x_{k+1} = x_k + v_k$$
$$r_k = 2x_k - 1$$

where v_k are i.i.d. Gaussian random variables. Suppose the LTL formula is given as $\phi = GF(r < 0)$, i.e., it is always (G) possible that in future (F), the requirement variable becomes negative. Then it can be verified [see Fig. 1(a)] that for any infinite sequence $r_0 r_1 \ldots r_m \ldots$ with $r_i \geq 0, \forall i \geq m$ (i.e., a sequence violating ϕ), any of its prefix has at least one infinite extension in which $(r_k < 0)$ is satisfied for infinitely many k (i.e., a sequence satisfying ϕ). Therefore G is not prediagnosable with respect to ϕ. In this case even with perfect observation $y_k = r_k$, the violation of ϕ cannot be detected.

Now consider the disturbance to be $v_k = sign(x_k)v'_k$, where v'_k is a positive-valued random variable, i.e., the noise v_k is dependent on the state variable x_k and is negative (resp., positive) if x_k is negative (resp., positive). As a result, the sequence $x_0 x_1 \ldots$, and also $r_0 r_1 \ldots$, are monotonically increasing (resp., decreasing) if x_0 is positive (resp., negative). Consider again the LTL formula $\phi = GF(r < 0)$. Then in this case, for every infinite sequence $r_0 r_1 \ldots r_m \ldots$ with $r_i \geq 0, \forall i \geq m$ (i.e., a sequence violating ϕ), there exists a finite prefix $r_0 \ldots r_k$ with $r_k \geq 0$ (so that $x_k = (r_k+1)/2 \geq 0.5$) whose all infinite extensions also violate ϕ. Then G is prediagnosable with respect to $GF(r < 0)$. In this case the Büchi automaton accepting $S_\phi \cap A_G$ is given in Fig. 1(b), where $\mathcal{L}_\phi = \{l_1\}$, i.e., $S_\phi \cap A_G$ is the limits of $(r < 0)^*$. The requirement model R is shown in Fig. 1(c), where the system behaviors satisfying ϕ visit l_1 infinitely often while those violating ϕ are absorbed at F. ■

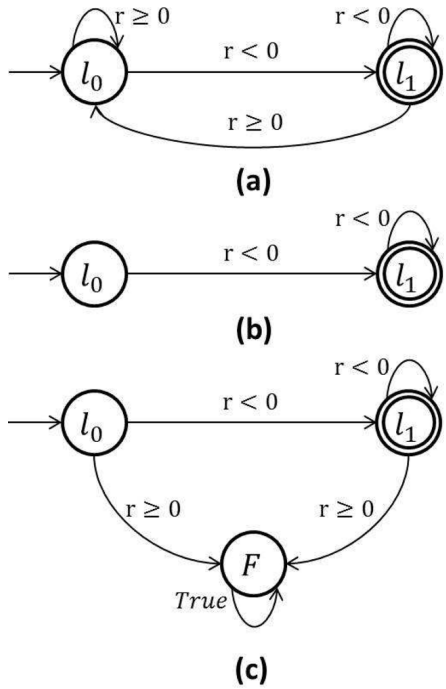


Fig. 1. (a) Büchi automaton for $\phi = GF(r < 0)$. (b) Büchi automaton for $S_\phi \cap A_G$ in the case of state dependent noise v_k. (c) The requirement model R for Example 1.

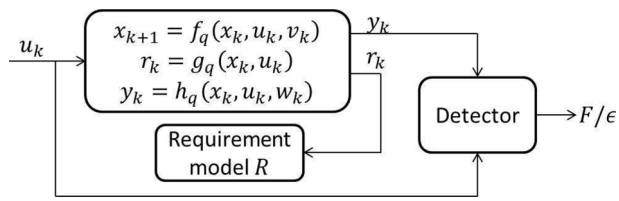


Fig. 2. Detection structure.

IV. Fault Detector Computation

Consider the detection structure of Fig. 2, where the monitored physical system G evolves according to stochastic difference (1)–(3), and requirement model R tracks its own discrete location as the requirements variable r_k evolves. At any given time, the true state of requirement model R is not available to the detector and must be estimated from observed history of inputs and outputs. We transform this problem of estimating requirement-violation to fault-location reachability estimation in

an I/O-SHA model that captures the behaviors of both G and R in a unified manner.

We first introduce the notion of an I/O-SHA, extending the logical inputI/O-HA given in [33].

A. Input–Output Stochastic Hybrid Automaton

Definition 2: An I/O-SHA is a 10-tuple $P = (L, X, U, Y, \Sigma, \Delta, \ell_0, d_0, L_m, E)$, where

- L is the set of locations (symbolic states), and each $l \in L$ is a 3-tuple $l = (G_l, f_l, h_l)$, where
  - $G_l : X \times U \to [0,1]$ is the location invariant probability satisfying (4) below;
  - $f_l : X \times U \times X \to [0,1]$ assigns for each $(x,u) \in X \times U$ a probability density function $f_l(\cdot|x,u)$ on data space X;
  - $h_l : X \times U \times Y \to [0,1]$ assigns for each $(x,u) \in X \times U$ a probability density function $h_l(\cdot|x,u)$ on output space Y.
- $X = X_1 \times \cdots \times X_n \subseteq \mathbb{R}^n$ is the set of data (numerical states), and hence the hybrid state space of P is given by $L \times X$;
- $U = U_1 \times \cdots \times U_m \subseteq \mathbb{R}^m$ is the set of numerical inputs;
- $Y = Y_1 \times \cdots \times Y_p \subseteq \mathbb{R}^p$ is the set of numerical outputs;
- Σ is the set of symbolic inputs;
- Δ is the set of symbolic outputs;
- $\ell_0 : L \to [0,1]$ is the initial probability distribution for the locations;
- $x_0 : X \to [0,1]$ is the initial probability distribution for the data values;
- $L_m \subseteq L$ is the set of final locations;
- E is the set of edges (transitions), and each $e \in E$ is a 7-tuple $e = (o_e, t_e, \sigma_e, \delta_e, G_e, f_e, h_e)$, where
  - $o_e \in L$ is the original location,
  - $t_e \in L$ is the terminal location,
  - $\sigma_e \in \Sigma \cup \{\epsilon\}$ is the symbolic input,
  - $\delta_e \in \Delta \cup \{\epsilon\}$ is the symbolic output,
  - $G_e : X \times U \to [0,1]$ is the guard probability satisfying (4) below,
  - $f_e : X \times U \times X \to [0,1]$ assigns for each $(x,u) \in X \times U$ a probability density function $f_e(\cdot|x,u)$ on data space X,
  - $h_e : X \times U \times Y \to [0,1]$ assigns for each $(x,u) \in X \times U$ a probability density function $h_e(\cdot|x,u)$ on output space Y.

Remark 3: In Definition 2, G_l and G_e, where $l \in L, e \in E$, capture the probabilities that an I/O-SHA stays in current location l or executes a transition e, and so it satisfies the following stochasticity constraint:

$$\forall (x,u) \in X \times U, \sigma \in \Sigma \cup \{\epsilon\}, \quad G_l(x,u) + \sum_{e \in E : \sigma_e = \sigma} G_e(x,u) \leq 1. \tag{4}$$

Note that, in a special setting, the range space of G_l and G_e can simply be the binary set $\{0,1\}$ [33], i.e., given any (x,u), an I/O-SHA will either stay at current location or execute one transition, with probability 1. Then, the guard/invariant can be equivalently written as logical predicates $\overline{G}_l := \{(x,u) : G_l(x,u) = 1\} \subseteq X \times U$ and $\overline{G}_e := \{(x,u) : G_e(x,u) = 1\} \subseteq X \times U$. Since in this paper we consider refinement of discrete-time stochastic systems against their logical LTL formula, only logical guards/invariants are needed in the refined I/O-SHA models. ■

An I/O-SHA P starts from an initial distribution ℓ_0 over L and an initial distribution x_0 over X. At each time step, given a current location l, current data value x and input value u, upon the arrival of a symbolic input $\sigma \in \Sigma \cup \{\epsilon\}$, P evolves either within current location with probability $G_l(x,u)$ or executes an outgoing edge e such that $\sigma_e = \sigma$ with probability $G_e(x,u)$. In the former case, it updates data variable x according to distribution $f_l(\cdot|x,u)$, and output variable y is assigned a value according to distribution $h_l(\cdot|x,u)$. In the latter case, distributions $f_e(\cdot|x,u)$ and $h_e(\cdot|x,u)$ are used for updating x and y, and a symbolic output δ_e is emitted.

Remark 4: In [26] and [29], the authors proposed discrete time stochastic hybrid systems (DTSHS), which includes hybrid state/control space. The I/O-SHA model introduced here is more general than the DTSHS model: state variables of a DTSHS are fully observed, whereas data variables of an I/O-SHA are only partially and unreliably observed. The notion of I/O-SHA can also be utilized to model cyberphysical systems [21], [40] where a cyber (discrete) component interacts with a physical (continuous) component. ■

B. Modeling the Refined System as a Stochastic I/O-HA

Next, we present the refinement of a system against its LTL formula that is translated into a requirement model R as described in Section III. Given a physical system G with dynamics described by (1)–(3) and the requirement model R, the refinement is modeled by an I/O-SHA G^R, where

- L is given by the state space of R, $\ell_0 = \delta(l_0^\phi)$ where δ is the Dirac delta function, and $L_m = \{F\}$;
- X, U, Y are given by the state/input/output space of G, respectively, and $\Sigma = \Delta = \emptyset$;
- the discrete transition structure of G^R is preserved from that of R;
- for each location $l \in L$:
  - location invariant $\overline{G}_l$ is given by $\overline{G}_l = \{(x,u) : g(x,u)$ violates the predicates over each outgoing transition from l in $R\}$,
  - probability density functions $f_l(\cdot|x,u)$ and $h_l(\cdot|x,u)$ for data updates and output assignments are determined by the distributions of v_k and w_k, together with the functions f and h of G,
- for each $e = (l, l', \sigma_e, \delta_e, \overline{G}_e, f_e, h_e)$, e is a transition of G^R (i.e., $e \in E$), if and only if
  - there exists a transition from l to l' in R,
  - $\overline{G}_e = \{(x,u) : g(x,u)$ satisfies the predicates over the above transition of $R\}$,
  - $\sigma_e = \delta_e = \epsilon$, $f_e(x_r|x,u) = \delta(x_r - x)$, and $h_e(\cdot|x,u)$ is the identity function that keeps output values unchanged on discrete transitions.

Remark 5: The refinement G^R captures the behaviors of both G and R in a unified manner such that any system behavior associated with an indicator transitions G^R to the fault-location $L_m = \{F\}$. ■

C. State Estimation for I/O-SHA

In order to aid the estimation of fault location reachability, we present the stochastic filtering equations to recursively estimate state distributions of I/O-SHA. Denote the history of observed inputs/outputs up to a time k as $u^k = u_0 \dots u_k$, $y^k = y_0 \dots y_k$ and let $z^k = (y^k, u^k)$. Define $\pi_{k+1}(\cdot|z^k) : L \to [0,1]$ as

$$\forall l \in L, \qquad \pi_{k+1}(l|z^k) := Pr(l_{k+1} = l|z^k)$$

which is the conditional probability distribution over discrete locations given the observations until time k. We further define two probability distribution functions over continuous variables of an I/O-SHA. The first one is the *prior* distribution $p_{k|k-1}(\cdot|z^{k-1}, l_k) : X \to [0,1]$ given by

$$\forall l_k \in L, x \in X, p_{k|k-1}(x|z^{k-1}, l_k) := p_{x_k|z^{k-1},l_k}(x|z^{k-1}, l_k)$$

which is the probability density function over continuous variables at time k, given z^{k-1} (i.e., prior to the input/output at time k) and l_k (the discrete location at time k). The second one is the *posterior* distribution $p_{k|k}(\cdot|z^k, l_k) : X \to [0,1]$ given by

$$\forall l_k \in L, x \in X, p_{k|k}(x|z^k, l_k) := p_{x_k|z^k,l_k}(x|z^k, l_k)$$

which is the probability density function over continuous variables at time k, given z^k (i.e., post to the input/output at time k) and l_k (the discrete location at time k).

The following equations initialize and recursively update the state distributions π_k, $p_{k|k}$ and $p_{k+1|k}$ for an I/O-SHA upon the arrival of the kth input/output pair for each $l \in L, x \in X$:

$$\pi_0(l|z^{-1}) = \ell_0(l) \tag{5}$$

$$p_{0|0}(x|z^0, l) = x_0(x) \tag{6}$$

$$p_{k|k}(x|z^k, l_k) = \frac{h_{l_k}(y_k|x, u_k) p_{k|k-1}(x|z^{k-1}, l_k)}{\int_D h_{l_k}(y_k|x_k, u_k) p_{k|k-1}(x_k|z^{k-1}, l_k) dx_k} \tag{7}$$

$$\pi_{k+1}(l|z^k) = \sum_{l_k \in L} \pi_k(l_k|z^{k-1}) \times \int_{D(l_k \to l|u_k)} p_{k|k}(x_k|z^k, l_k) dx_k \tag{8}$$

$$p_{k+1|k}(d|z^k, l_{k+1}) = \frac{1}{\pi_{k+1}(l_{k+1}|z^k)} \sum_{l_k} \pi_k(l_k|z^{k-1}) \times \int_{D(l_k \to l_{k+1}|u_k)} f_{l_{k+1}}(x|x_k, u_k) \times p_{k|k}(x_k|z^k, l_k) dx_k, \tag{9}$$

where $D(l_i \to l_j|u_i) \subseteq X$ for each l_i, l_j and u_i is defined as $D(l_i \to l_j|u_i) := \{x_i \in X : \exists e \in E, o_e = l_i, t_e = l_j, (u_i, x_i) \in \overline{G}_e\}$, i.e., it is the set of data values that enable the edge from l_i to l_j when the input is u_i. The detailed derivations of (7)–(9) are given in the Appendix.

D. Detection Statistics and Detection Scheme

Now that we have computed the state probability distributions given the input/output sequence up to a current time k, we can use this to compute the *likelihood of fault*, which is the probability of the refinement G^R being at the fault-location $L_m = \{F\}$, and is given by

$$P_F^k := \sum_{l \in L_m} \pi_{k+1}(l|z^k). \tag{10}$$

Note that P_F^k can be found by first computing π_k, which in turn is computed by the filter (5)–(9). A detector issues a fault decision "F" whenever this likelihood of fault is higher than a threshold, i.e., when $P_F^k > \rho$, and remains silent otherwise. The detector $\mathcal{D} : (U \times Y)^{\mathbb{N}} \to \{F, \epsilon\}$ is formally defined as

$$\forall z^k \in (U \times Y)^{\mathbb{N}}, [\mathcal{D}(z^k) = F] \Leftrightarrow [\exists j \le k, P_F^j > \rho]. \tag{11}$$

Note that once the detector issues F, it issues F for all subsequent steps, i.e., the detector "doesn't change its mind". This is a desired property of a detector since it is designed to detect the execution of an indicator that occurred in the past.

Remark 6: Note that performing filter (5)–(9) requires complexity that is linear to the size of discrete locations and quadratic to the number of sample points in X, while computing P_F^k in (10) needs complexity linear to the size of L.

Note also that, while we only consider discrete-time stochastic systems with single mode of dynamics, the framework can be straightforwardly extended to the case where the system under diagnosis is itself an I/O-SHA. In this case, the locations of the refinement G^R are given by the location-pairs of G and R, and the guards/invariants are given by intersections of guards/invariants in G and R. The detection algorithm (5)–(11) continues to apply to this more general setting where G itself is an I/O-SHA. ■

Remark 7: In this paper, we consider a fault to be a violation of given LTL formulas. As studied in literature [15], [16], a fault may be modeled as a change in system dynamics. We can subsume this situation in our framework by considering the refinement G^R in which the probability density functions $f_l(\cdot|x, u)$ for location $l = F$ undergoes a dynamics change due to the occurrence of fault. Then the fault detection problem is again reduced to fault-location reachability detection problem for G^R, i.e., estimation of the probability of reaching fault-location F, which can be solved by our proposed algorithm (5)–(11). ■

V. Case Study: A Room-Heating Problem

Here, we illustrate the fault detection computations described above by applying to a room heating benchmark, which aims to regulate the temperature in a single room with a single heater and is inspired from [34]. Let the continuous variable x_k present the room temperature at time k, and the binary variable u_k denote the status of the heater, with $u_k = 1$ if the heater is on at time k and 0 otherwise. The room temperature x_k is assumed to evolve according to the linear stochastic difference equation

$$x_{k+1} = x_k + a(x_a - x_k) + bu_k + v_k$$

and the requirement and output variables are given by

$$r_k = \begin{bmatrix} u_k \\ x_k \end{bmatrix},$$
$$y_k = x_k + w_k$$

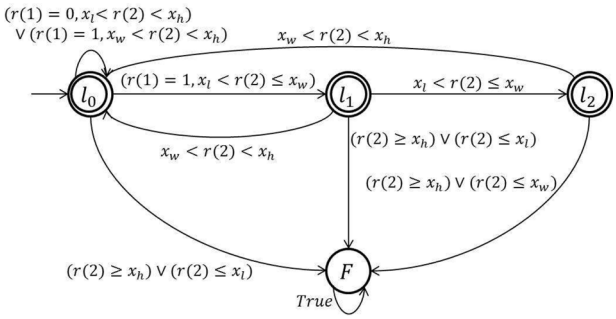


Fig. 3. Rrequirement model R for single-room heating problem.

TABLE I
LIST OF $D(l_i \to l_j|u)$

$D(l_0 \to l_0\|u = 0)$	(x_l, x_h)
$D(l_0 \to l_0\|u = 1)$	(x_w, x_h)
$D(l_0 \to l_1\|u = 1)$	$(x_l, x_w]$
$D(l_1 \to l_0\|u \in \{0,1\})$	(x_w, x_h)
$D(l_1 \to l_2\|u \in \{0,1\})$	$(x_l, x_w]$
$D(l_2 \to l_0\|u \in \{0,1\})$	(x_w, x_h)
$D(l_0 \to F\|u \in \{0,1\})$	$(-\infty, x_l] \cup [x_h, \infty)$
$D(l_1 \to F\|u \in \{0,1\})$	$(-\infty, x_l] \cup [x_h, \infty)$
$D(l_2 \to F\|u \in \{0,1\})$	$(-\infty, x_w] \cup [x_h, \infty)$
$D(F \to F\|u \in \{0,1\})$	$(-\infty, \infty)$
Others	$\emptyset$

where x_a is the (constant) ambient temperature, and the disturbance v_k and the noise w_k are zero mean Gaussian random variables with variances σ_v^2 and σ_w^2, respectively.

For safety purposes, it is required that the room temperature satisfies $x_l \leq x_k \leq x_h$ for all k. It is also required that the room temperature is guaranteed to be higher than x_w in at most 2 steps after the heater is turned on. Note $x_h > x_w > x_l$ are constants, specified by user/designer. Such correctness requirement can be expressed as LTL formula ϕ

$$\begin{aligned}\phi = G[\{x_l < r(2) < x_h\} \\ \wedge \{(r(1) = 1) \Rightarrow (r(2) > x_w) \vee X(r(2) > x_w) \\ \vee XX(r(2) > x_w)\}]. \quad (12)\end{aligned}$$

It can be verified that the aforementioned system is prediagnosable with respect to ϕ, and the requirement model R is shown in Fig. 3, which has four states and nine edges, while reaching the state F indicates the violation of formula (12).

The refinement G^R is such that $L = \{l_0, l_1, l_2, F\}$, $U = \{0, 1\}$, $X = Y = \mathbb{R}$, $\ell_0 = \delta(l_0)$, $x_0 = \delta(x)$, $L_m = \{F\}$ and the edges are as shown in Fig. 3. For each $l \in L$,

$$\begin{aligned}f_l(\cdot|x, u) &= \mathcal{N}(\cdot|x + a(x_a - x) + bu, \sigma_v^2) \\ h_l(\cdot|x, u) &= \mathcal{N}(\cdot|x, \sigma_w^2)\end{aligned}$$

where $\mathcal{N}(\cdot|\mu, \sigma^2)$ denotes Gaussian distribution with mean μ and variance σ^2. Recall that, as defined in Section IV-C, $D(l_i \to l_j|u_i)$ for each l_i, l_j and u_i is the set of data values that enable the edge from l_i to l_j when the input is u_i. For each $l_j, l_j \in L$ and $u \in U$, $D(l_i \to l_j|u)$ can be easily computed and is shown in Table I.

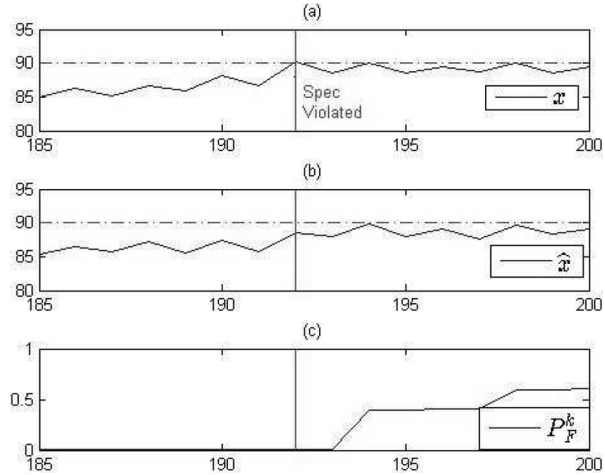


Fig. 4. Detection result for a run that violates the correctness requirement by exceeding upper limit x_h. (a) True $r(2) = x$. (b) Estimation $\widehat{x}$ of x. (c) Likelihood of fault computed by detector.

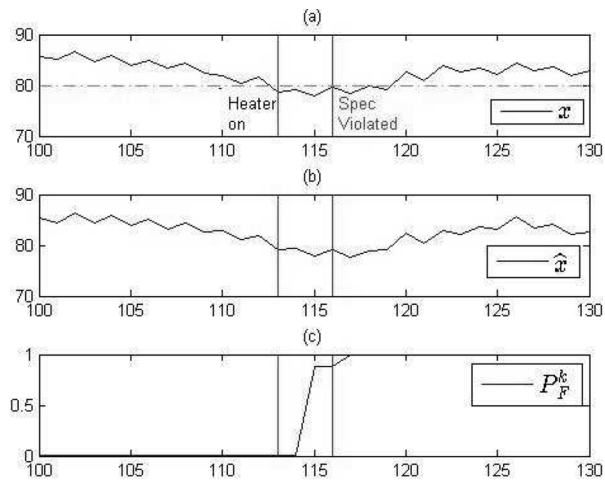


Fig. 5. Detection result for a run that violates the correctness requirement by failing to reach x_w within two steps after the heater is on. (a) True $r(2) = x$. (b) Estimation $\widehat{x}$ of x. (c) Likelihood of fault computed by detector.

For the computational study, we set $x_a = 70$, $a = 0.1, b = 3, \sigma_v^2 = \sigma_w^2 = 0.4$, and suppose the system is initialized at $x_0 = 80$ and l_0. Suppose the specification parameters are $x_l = 70$, $x_h = 90$ and $x_w = 80$. For simulation, the continuous space is discretized by a grid size of 0.1 over the range $[65, 100]$. The input is such that the heater switches between on and off at each discrete time.

A total of 5000 runs, with terminal time $T = 200$, were simulated, out of which there were 457 runs violating ϕ. We implemented the detection algorithm (5)–(11), and the results are shown in Figs. 4–6. In Fig. 4, the room temperature exceeds the upper limit, whereas in Fig. 5, ϕ is violated since the room temperature remains below $x_w = 80$ two steps after the heater is on. In both cases, the likelihood of fault, P_F, arises soon after the specification is violated, and the fault can be detected with a delay of 7 steps by using a detection threshold $\rho = 0.5$. The performance of the detection scheme can be evaluated by the errors in terms of false alarms and missed detections (formally defined in next section), and Fig. 6 shows the number of runs that are false-alarmed or missed-detected over the 5000 runs, as

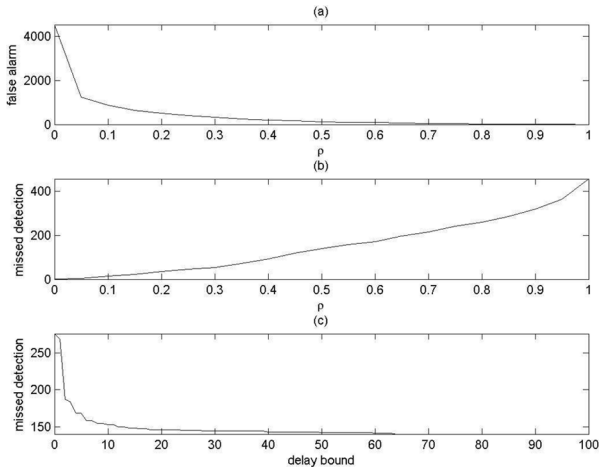


Fig. 6. (a) Number of false alarms as a function of threshold. (b) Number of missed detections as a function of threshold. (c) Number of missed detections as a function of detection delay, when the threshold is $\rho = 0.5$.

detection threshold ρ and detection delay n are changed. The number of runs that are false-alarmed is a function of detection threshold and decreases as detection threshold increases, while the number of runs that are missed-detected is a function of both detection threshold and detection delay. When detection delay is fixed, the number of runs that are missed-detected increases as detection threshold increases, whereas it decreases as the detection delay increases while the detection threshold is fixed.

VI. Performance Evaluation and Stochastic Diagnosability

As illustrated in the case study in the previous section, the performance of the detection scheme proposed above can be measured in terms of false alarm (FA) and missed detection (MD) rates. Here, we formally define FA and MD rates, by first introducing the following notions.

A finite run of the system is a finite execution of the stochastic difference (1)–(3), denoted as $\overline{z} := (u^{|\overline{z}|}, x^{|\overline{z}|}, r^{|\overline{z}|}, y^{|\overline{z}|})$, where $|\overline{z}| < \infty$ and for each $o \in \{u, x, r, y\}$, $o^{|\overline{z}|} := o_0 \dots o_{|\overline{z}|}$. A run is a fault-run if the associated sequence of requirement variables $r^{|\overline{z}|}$ is an indicator, i.e., $r^{|\overline{z}|} \in I_\phi(G)$, where recall that $I_\phi(G)$ is the set of all indicators. A run is a nonfault-run if it is not a fault-run. Given two runs $\overline{z}_1 := (u_1^{|\overline{z}_1|}, x_1^{|\overline{z}_1|}, r_1^{|\overline{z}_1|}, y_1^{|\overline{z}_1|})$ and $\overline{z}_2 := (u_2^{|\overline{z}_2|}, x_2^{|\overline{z}_2|}, r_2^{|\overline{z}_2|}, y_2^{|\overline{z}_2|})$, $\overline{z}_1$ is said to be a prefix of $\overline{z}_2$, denoted as $\overline{z}_1 \leq \overline{z}_2$, if $|\overline{z}_1| \leq |\overline{z}_2|$ and $o_2^{|\overline{z}_1|} \equiv o_1^{|\overline{z}_1|}$ for each $o \in \{u, x, r, y\}$. In this case, we denote $\overline{z}_2 \backslash \overline{z}_1$ as an extension of $\overline{z}_1$.

Associated with each run $\overline{z}$ is a sequence of detection statistics, $P_F^0, P_F^1, \dots, P_F^{|\overline{z}|}$, computed using (5)–(10). Then a FA occurs if the detector issues F decision for a nonfault-run, and so FA rate can be defined as

$$P^{fa} := Pr(\overline{z} : r^{|\overline{z}|} \notin I_\phi(G) \wedge P_F^{|\overline{z}|} > \rho). \tag{13}$$

A MD occurs if the detector remains silent n steps after the system executes an indicator, where n is the detection delay bound allowed by the detector. So MD rate can be defined as

$$P^{md} := Pr(\overline{z} : \exists k < |\overline{z}| - n, r^k \in I_\phi(G), P_F^{|\overline{z}|} \leq \rho). \tag{14}$$

Next we present a characterization of the class of systems for which detectors with arbitrary accuracies can be designed, by introducing the notion of *Stochastic-Diagnosability* which requires that for any tolerable threshold ρ and error bound τ, there must exist a delay bound n such that for any fault-run, its extensions, longer than n and having likelihood of fault lower than ρ, occur with probability at most τ.

Definition 3: Given a system G subjected to an input-sequence drawn from a distribution μ, with correctness requirement expressed in LTL formula ϕ, (G, μ, ϕ) is said to be *Stochastically-Diagnosable*, or simply *S-Diagnosable*, if $\forall 1 > \rho, \tau > 0$, $\exists n \in \mathbb{N}$, such that for any fault-run $\overline{z}_0$

$$Pr(\overline{z} \backslash \overline{z}_0 : |\overline{z}| - |\overline{z}_0| > n, P_F^{|\overline{z}|} \leq \rho) < \tau. \tag{15}$$

The following theorem establishes the significance of the S-Diagnosability property, by showing its necessity and sufficiency for the existence of a detector to achieve any desired level of accuracy as measured in terms of FA and MD rates.

Theorem 1: For any FA rate $\nu > 0$ and MD rate $\tau > 0$, there exists a detection threshold ρ and delay bound n so that the rates of FA and MD defined by (13)–(14) satisfy $P^{fa} < \nu$ and $P^{md} < \tau$ if and only if (G, μ, ϕ) is S-Diagnosable.

Proof :

(Sufficiency) As shown in (13), for $\rho_1 > \rho_2 > 0$, $\{\overline{z} : r^{|\overline{z}|} \notin I_\phi(G) \wedge P_F^{|\overline{z}|} > \rho_1\} \subseteq \{\overline{z} : r^{|\overline{z}|} \notin I_\phi(G) \wedge P_F^{|\overline{z}|} > \rho_2\}$, and so FA rate decreases as detection threshold gets higher. Therefore, any FA rate ν can be achieved by adequately lowering the detection threshold. Let ρ_ν be the threshold that ensures FA rate ν. When (G, μ, ϕ) is S-Diagnosable, there exists an integer $n \in \mathbb{N}$ such that (15) holds. Therefore

$$\begin{aligned} P^{md} &= Pr(\overline{z} : \exists k < |\overline{z}| - n, r^k \in I_\phi(G), P_F^{|\overline{z}|} \leq \rho_\nu) \\ &= \sum_{\overline{z}_0 : r^{|\overline{z}_0|} \in I_\phi(G)} Pr(\overline{z}_0) \\ &\qquad \times Pr(\overline{z} \backslash \overline{z}_0 : |\overline{z}| - |\overline{z}_0| > n, P_F^{|\overline{z}|} \leq \rho_\nu) \\ &< \sum_{\overline{z}_0 : r^{|\overline{z}_0|} \in I_\phi(G)} Pr(\overline{z}_0)\tau < \tau. \end{aligned}$$

Thus, the sufficiency holds.

(Necessity) When (G, μ, ϕ) is not S-Diagnosable, there exists $\rho_0, \tau_0 > 0$ and a fault-run $\overline{z}_0$ such that for any $n \in \mathbb{N}$, (15) does not hold, i.e.,

$$Pr(\overline{z} \backslash \overline{z}_0 : |\overline{z}| - |\overline{z}_0| > n, P_F^{|\overline{z}|} \leq \rho_0) \geq \tau_0. \tag{16}$$

Let $\nu > 0$ be such that $\rho_\nu = \rho_0$. Then, for any $n \in \mathbb{N}$,

$$\begin{aligned} P^{md} &= Pr(\overline{z} : \exists k < |\overline{z}| - n, r^k \in I_\phi(G), P_F^{|\overline{z}|} \leq \rho_0) \\ &\geq Pr(\overline{z}_0) Pr(\overline{z} \backslash \overline{z}_0 : |\overline{z}| - |\overline{z}_0| > n, P_F^{|\overline{z}|} \leq \rho_0) \\ &\geq Pr(\overline{z}_0)\tau_0 =: \tau_{low}. \end{aligned}$$

Therefore in this case, a MD rate of τ_{low} can not be achieved. Thus the necessity holds. ■

Remark 8: Theorem 1 identifies the class of systems for which a detector of any desired accuracy can be constructed.

Therefore, the S-Diagnosability property should be checked before designing a detector—a desired accuracy may not be achievable if S-Diagnosability is not satisfied. Verifying the S-Diagnosability property is an open problem, subject for future work, along with an algorithm that computes a detector so as to ensure the desired rates of FA and MD. ■

Example 2: Let us revisit the second system in Example 1. The state equation is given by $x_{k+1} = x_k + v_k$, where the disturbance $v_k = sign(x_k)v'_k$ and v'_k is a positive-valued random variable with density function $f_{v'}$ whose mean is 1. The requirement and output variables are given by $r_k = 2x_k - 1$ and $y_k = 2x_k - 1 + w_k$, where w_k are i.i.d. zero mean Gaussian random variables with variance σ_w. Consider again the LTL formula $\phi = GF(r < 0)$. As shown in Example 1, the system is prediagnosable with respect to ϕ. Moreover, according to Fig. 1(c), detecting the requirement-violation by time k is equivalent to detecting the existence of $l \leq k$ such that $r_l \geq 0$ (or $x_k \geq 0.5$).

Now consider a fault-run $\overline{z}_0$ and its extension $\overline{z}\backslash\overline{z}_0$, we have

$$\begin{aligned} P_F^{|\overline{z}|} &= 1 - Pr(\forall 0 \leq l \leq |\overline{z}|, x_l < 0.5 \mid y_0, \ldots, y_{|z|}) \\ &= 1 - \int_{-\infty}^{0.5} \cdots \int_{-\infty}^{0.5} \mathcal{N}(y_{|\overline{z}|} \mid 2x_{|\overline{z}|} - 1, \sigma_w) f(x_{|\overline{z}|} - x_{|\overline{z}|-1}) \\ &\quad \times \cdots \times \mathcal{N}(y_1 \mid 2x_1 - 1, \sigma_w) f(x_1 - x_0) \\ &\quad \times \mathcal{N}(y_0 \mid 2x_0 - 1, \sigma_w) d(x_0) dx_0 \cdots dx_{|\overline{z}|} \\ &\geq 1 - \int_{-\infty}^{0.5} \mathcal{N}(y_{|\overline{z}|} \mid 2x_{|\overline{z}|} - 1, \sigma_w) dx_{|\overline{z}|}. \end{aligned}$$

For any $1 > \rho > 0$, define y_ρ be such that

$$\int_{-\infty}^{0.5} \mathcal{N}(y_\rho \mid 2x_{|\overline{z}|} - 1, \sigma_w) dx_{|\overline{z}|} = 1 - \rho.$$

Then $(y_{|\overline{z}|} > y_\rho) \Rightarrow (P_F^{|\overline{z}|} > \rho)$, and so $(P_F^{|\overline{z}|} \leq \rho) \Rightarrow (y_{|\overline{z}|} \leq y_\rho)$. Hence,

$$Pr(\overline{z}\backslash\overline{z}_0 : P_F^{|\overline{z}|} \leq \rho) \leq Pr(\overline{z}\backslash\overline{z}_0 : y_{|\overline{z}|} \leq y_\rho).$$

According to the discussion of Example 1, for any fault-run $\overline{z}_0$, the sequence of state variables $x_0 x_1 \ldots$ is monotonically increasing with average increase of 1 at each time step. Therefore $\lim_{|\overline{z}| \to \infty} x_{|\overline{z}|} = \infty$ and so for a fixed ρ (or y_ρ), $\lim_{|\overline{z}| \to \infty} Pr(y_{|\overline{z}|} < y_\rho) = 0$ (See Fig. 7). Then we have $\lim_{n \to \infty} Pr(\overline{z}\backslash\overline{z}_0 : |\overline{z}| - |\overline{z}_0| > n, y_{|\overline{z}|} < y_\rho) = 0$, i.e., for any $\tau > 0$, there exists $n \in \mathbb{N}$, such that

$$\begin{aligned} &Pr(\overline{z}\backslash\overline{z}_0 : |\overline{z}| - |\overline{z}_0| > n, P_F^{|\overline{z}|} \leq \rho) \\ &\leq Pr(\overline{z}\backslash\overline{z}_0 : |\overline{z}| - |\overline{z}_0| > n, y_{|\overline{z}|} \leq y_\rho) \\ &< \tau. \end{aligned}$$

Since the above analysis works for any $\rho > 0$, one can conclude that S-Diagnosability holds in this example. According to Theorem 1, any desired rates of FA and MD can be achieved by suitably choosing threshold and delay bound. When the FA rate ν is made tighter by decreasing it, a larger detection threshold ρ is required, while when the MD rate τ is made tighter by lowering it, a detector needs to wait for a longer delay bound n.

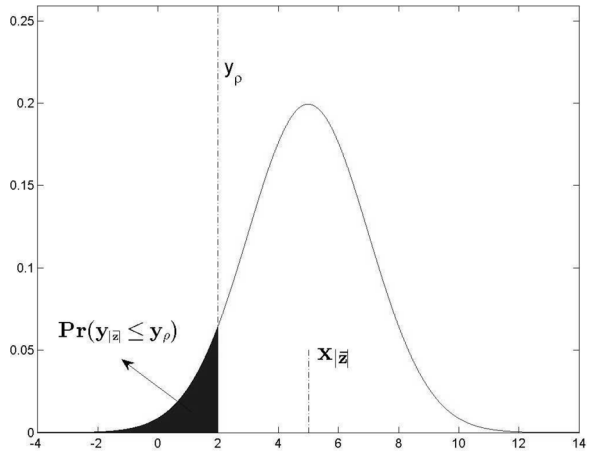


Fig. 7. Gaussian distribution with mean $x_{|\overline{z}|}$ and variance σ_w.

VII. Conclusion

This paper studied the fault detection of discrete-time stochastic systems subject to linear-time temporal logic correctness requirement. The continuous physical system (modeled as stochastic difference equations) was refined against its LTL correctness requirement to yield an input-output stochastic hybrid automaton which preserves the behavior of the physical system and captures requirement-violation as a reachability property to a fault-location. The likelihood of fault was proposed as a detection statistic, and was recursively computed for issuing a detection decision (a fault decision is issued when the likelihood of fault arises above a suitably chosen threshold, implying the likelihood of fault has become "high" and so a fault is concluded). Although in the proposed framework, a fault is defined to be a violation of certain correctness requirement and does not necessarily result in a dynamics change, the framework can be straightforwardly adopted to capture fault models which involve a change in system dynamics as in [15], [16]. The proposed diagnosis procedure was implemented for a benchmark room heating problem to show its validity and applicability. The performance of the procedure was evaluated in terms of false alarm and missed detection rates, and the existence of detector for achieving any desired false alarm and missed detection rates was captured as Stochastic-Diagnosability introduced in this paper. As part of future work, analytical computation of the rates of false alarm and missed detection will be investigated. Also the verification of the Stochastic-Diagnosability property will be developed.

Appendix

Here we derive (7)–(9). According to the definition, we have

$$\begin{aligned} p_{k+1|k}(x|z^k, l_{k+1}) &= \frac{Pr(x_{k+1} = x, z^k, l_{k+1})}{Pr(z^k, l_{k+1})} \\ p_{k|k}(x|z^k, l_k) &= \frac{Pr(x_k = x, z^k, l_k)}{Pr(z^k, l_k)} \\ \pi_{k+1}(l|z^k) &= \frac{Pr(l_{k+1} = l, z^k)}{\sum_{l \in L} Pr(l_{k+1} = l, z^k)}. \end{aligned}$$

Therefore, we have

$$p_{k|k-1}(x|z^{k-1},l_k)=\frac{Pr(x_k=x,z^{k-1},l_k)}{Pr(z^{k-1},l_k)}$$
$$\pi_k(l|z^{k-1})=\frac{Pr(l_k=l,z^{k-1})}{\sum_{l\in L}Pr(l_k=l,z^{k-1})}.$$

Combining $p_{k|k}(x|z^k,l_k)$ and $p_{k|k-1}(x|z^{k-1},l_k)$, we obtain

$$\begin{aligned}
&p_{k|k}(x|z^k,l_k)\\
&\quad=\frac{Pr(x_k=x,z^k,l_k)}{Pr(z^k,l_k)}\\
&\quad=\frac{Pr(x_k=x,z^{k-1},l_k,(u_k,y_k))}{Pr(z^{k-1},l_k,(u_k,y_k))}\\
&\quad=\frac{Pr(x_k=x,z^{k-1},l_k)Pr(y_k|x_k=x,u_k,l_k)}{\sum_{x_k\in X}Pr(x_k,z^{k-1},l_k)Pr(y_k|x_k,u_k,l_k)}\\
&\quad=\frac{\frac{Pr(x_k=x,z^{k-1},l_k)}{Pr(z^{k-1},l_k)}Pr(y_k|x_k=x,u_k,l_k)}{\sum_{x_k\in X}\frac{Pr(x_k,z^{k-1},l_k)}{Pr(z^{k-1},l_k)}Pr(y_k|x_k,u_k,l_k)}\\
&\quad=\frac{p_{k|k-1}(x|z^{k-1},l_k)Pr(y_k|x_k=x,u_k,l_k)}{\sum_{x_k\in X}p_{k|k-1}(x_k|z^{k-1},l_k)Pr(y_k|x_k,u_k,l_k)}\\
&\quad=\frac{p_{k|k-1}(x|z^{k-1},l_k)h_{l_k}(y_k|x,u_k)}{\int_X p_{k|k-1}(x_k|z^{k-1},l_k)h_{l_k}(y_k|x_k,u_k)dx_k}
\end{aligned}$$

i.e.,

$$p_{k|k}(x|z^k,l_k)=\frac{p_{k|k-1}(x|z^{k-1},l_k)h_{l_k}(y_k|x,u_k)}{\int_X p_{k|k-1}(x_k|z^{k-1},l_k)h_{l_k}(y_k|x_k,u_k)dx_k}.$$

Thus, we have shown (7). Next by combining $\pi_{k+1}(l|z^k)$, $\pi_k(l|z^{k-1})$ and $p_{k|k}(x|z^k,l_k)$, we have

$$\begin{aligned}
&\pi_{k+1}(l|z^k)\\
&\quad=\frac{Pr(l_{k+1}=l,z^k)}{\sum_{l\in L}Pr(l_{k+1}=l,z^k)}\\
&\quad=Pr(l_{k+1}=l|z^k)\\
&\quad=\sum_{l_k\in L}\sum_{x_k\in D}Pr(l_k,l_{k+1}=l,x_k|z^k)\\
&\quad=\sum_{l_k\in L}\sum_{x_k\in D}Pr(l_{k+1}=l,x_k|l_k,z^k)Pr(l_k|z^k)\\
&\quad=\sum_{l_k\in L}\sum_{x_k\in D(l_k\to l|u_k)}Pr(x_k|l_k,z^k)Pr(l_k|z^{k-1})\\
&\quad=\sum_{l_k\in L}\pi_k(l_k|z^{k-1})\int_{D(l_k\to l|u_k)}p_{k|k}(x_k|z^k,l_k)dx_k.
\end{aligned}$$

Thus, we have established (8). Finally combining $p_{k+1|k}(x|z^k,l_{k+1})$, $\pi_{k+1}(l|z^k)$, $\pi_k(l|z^{k-1})$ and $p_{k|k}(x|z^k,l_k)$ yields

$$\begin{aligned}
&p_{k+1|k}(x|z^k,l_{k+1})\\
&\quad=\frac{Pr(x_{k+1}=x,z^k,l_{k+1})}{Pr(z^k,l_{k+1})}\\
&\quad=\frac{Pr(x_{k+1}=x,l_{k+1}|z^k)Pr(z^k)}{Pr(z^k,l_{k+1})}\\
&\quad=\frac{1}{\pi_{k+1}(l_{k+1}|z^k)}Pr(x_{k+1}=x,l_{k+1}|z^k)\\
&\quad=\frac{1}{\pi_{k+1}(l_{k+1}|z^k)}\sum_{l_k\in L}\sum_{x_k\in X}Pr(x_{k+1}=x,x_k,l_k,l_{k+1}|z^k)\\
&\quad=\frac{1}{\pi_{k+1}(l_{k+1}|z^k)}\sum_{l_k\in L}\sum_{x_k\in X}\\
&Pr(x_{k+1}=x,x_k,l_{k+1}|l_k,z^k)Pr(l_k|z^k)\\
&\quad=\frac{1}{\pi_{k+1}(l_{k+1}|z^k)}\sum_{l_k\in L}\sum_{x_k\in X}Pr(l_k|z^{k-1})\\
&Pr(x_{k+1}=x,|x_k,l_{k+1},l_k,z^k)Pr(x_k,l_{k+1}|l_k,z^k)\\
&\quad=\frac{1}{\pi_{k+1}(l_{k+1}|z^k)}\sum_{l_k\in L}\pi_k(l_k|z^{k-1})\sum_{x_k\in D(l_k\to l_{k+1}|u_k)}\\
&Pr(x_{k+1}=x,|x_k,l_{k+1},l_k,z^k)Pr(x_k,l_{k+1}|l_k,z^k)\\
&\quad=\frac{1}{\pi_{k+1}(l_{k+1}|z^k)}\sum_{l_k\in L}\pi_k(l_k|z^{k-1})\sum_{x_k\in D(l_k\to l_{k+1}|u_k)}\\
&f_{l_{k+1}}(x|x_k,u_k)Pr(x_k|l_k,z^k)\\
&\quad=\frac{1}{\pi_{k+1}(l_{k+1}|z^k)}\sum_{l_k\in L}\pi_k(l_k|z^{k-1})\\
&\qquad\int_{D(l_k\to l_{k+1}|u_k)}f_{l_{k+1}}(x|x_k,u_k)p_{k|k}(x_k|z^k,l_k)dx_k
\end{aligned}$$

i.e.,

$$\begin{aligned}
p_{k+1|k}(x|z^k,l_{k+1})&=\frac{1}{\pi_{k+1}(l_{k+1}|z^k)}\sum_{l_k\in L}\pi_k(l_k|z^{k-1})\\
&\int_{D(l_k\to l_{k+1}|u_k)}f_{l_{k+1}}(x|x_k,u_k)p_{k|k}(x_k|z^k,l_k)dx_k.
\end{aligned}$$

Thus, we have also established (9).

REFERENCES

[1] C. Zhou, R. Kumar, and S. Jiang, "Keynote: Hierarchical fault detection in embedded control software," in *Proc. IEEE Int. Comput. Software Applicant. Conf.*, Jul. 2008, pp. 816–823.

[2] R. Isermann, R. Schwarz, and S. Stolzl, "Fault-tolerant drive-by-wire systems," *IEEE Control Syst. Mag.*, vol. 27, no. 5, pp. 64–81, Oct. 2002.

[3] M. He and J. Zhang, "A dependency graph approach for fault detection and localization towards secure smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 342–351, Jun. 2011.

[4] K. Kim and E. B. Bartlett, "Nuclear power plant fault diagnosis using neural networks with error estimation by series association," *IEEE Trans. Nucl. Sci.*, vol. 43, no. 4, pp. 2373–2388, Aug. 1996.

[5] C. Favre, "Fly-by-wire for commercial aircraft: the airbus experience," *Int. J. Control*, vol. 59, no. 1, pp. 139–157, 1994.

[6] G. Westerman, R. Kumar, C. Stroud, and J. Heath, "Discrete event system approach for delay fault analysis in digital circuits," in *Proc. 1998 Amer. Control Conf.*, Philadelphia, PA, USA, Jun. 1998, pp. 239–243.

[7] I. Hwang, S. Kim, Y. Kim, and C. E. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 3, pp. 636–653, May 2010.

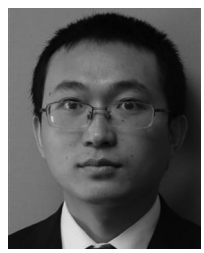
[8] M. E. Basseville and I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993.

[9] J. Chen and R. Kumar, "Online failure diagnosis of stochastic discrete event systems," in *Proc. IEEE Multi-Conf. Syst. and Control*, Hyderabad, India, Aug. 2013, pp. 194–199.

[10] J. Chen and R. Kumar, "Failure detection framework for stochastic discrete event systems with guaranteed error bounds," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1542–1553, Jun. 2015.

[11] J. Chen and R. Kumar, "Failure diagnosis of discrete-time stochastic systems subject to temporal logic correctness requirements," in *Proc. IEEE Int. Conf. Netw. Sensing, and Control*, Miami, FL, USA, Apr. 2014, pp. 42–47.

[12] D. Thorsley and D. Teneketzis, "Diagnosability of stochastic discrete-event systems," *IEEE Trans. Autom. Control*, vol. 50, no. 4, pp. 476–492, Apr. 2005.
[13] J. Chen and R. Kumar, "Polynomial test for stochastic diagnosability of discrete event systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 4, pp. 969–979, Oct. 2013.
[14] J. Chen and R. Kumar, "Decentralized failure diagnosis of stochastic discrete event systems," in *Proc. 9th IEEE Int. Conf. Autom. Sci. Eng.*, Madison, WI, USA, Aug. 2013, pp. 1083–1088.
[15] M. Zhong, S. X. Ding, J. Lam, and H. Wang, "An LMI approach to design robust fault detection filter for uncertain lti systems," *Automatica*, vol. 39, no. 3, pp. 543–550, 2003.
[16] R. H. Chen, D. L. Mingori, and J. L. Speyer, "Optimal stochastic fault detection filter," *Automatica*, vol. 39, no. 3, pp. 377–390, Mar. 2003.
[17] U. Lerner, R. Parr, D. Koller, and G. Biswas, "Bayesian fault detection and diagnosis in dynamic systems," in *Proc. Nat. Conf. Artific. Intell.*, Austin, TX, USA, Aug. 2000, pp. 531–537.
[18] F. Zhao, X. Koutsoukos, H. Haussecker, J. Reich, and P. Cheung, "Monitoring and fault diagnosis of hybrid systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 6, pp. 1225–1240, Dec. 2005.
[19] A. Bauer, M. Leucker, and C. Schallhart, "Comparing LTL semantics for runtime verification," *J. Logic Computation*, vol. 20, no. 3, pp. 651–674, 2010.
[20] A. Bauer, M. Leucker, and C. Schallhart, "Runtime verification for LTL and TLTL," *ACM Trans. Software Eng. Methodol.*, vol. 20, no. 4, p. 14, 2011.
[21] A. P. Sistla, M. Zefran, and Y. Feng, "Runtime monitoring of stochastic cyber-physical systems with hybrid state," *Lecture Notes in Comput. Sci.*, vol. 7186, pp. 276–293, 2012.
[22] S. Jiang and R. Kumar, "Failure diagnosis of discrete-event systems with linear-time temporal logic specifications," *IEEE Trans. Autom. Control*, vol. 49, no. 6, pp. 934–945, Jun. 2004.
[23] S. Jiang and R. Kumar, "Diagnosis of repeated failures for discrete event systems with linear-time temporal-logic specifications," *IEEE Trans. Autom. Sci. Eng.*, vol. 3, no. 1, pp. 47–59, Jan. 2006.
[24] B. Yordanov, J. Tumova, I. Cerna, J. Barnat, and C. Belta, "Temporal logic control of discrete-time piecewise affine systems," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1491–1504, Jun. 2012.
[25] G. E. Fainekos, S. G. Loizou, and G. J. Pappas, "Translating temporal logic to controller specifications," in *Proc. 45th IEEE Conf. Decision Control*, San Diego, CA, USA, Dec. 2006, pp. 899–904.
[26] S. Summers and J. Lygeros, "Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem," *Automatica*, vol. 46, no. 12, pp. 1951–1961, Sep. 2010.
[27] A. Abate, A. D. Innocenzo, and M. D. D. Benedetto, "Approximate abstractions of stochastic hybrid systems," *IEEE Trans. Autom. Control*, vol. 56, no. 11, pp. 2688–2694, Nov. 2011.
[28] A. A. Julius and G. J. Pappas, "Approximations of stochastic hybrid systems," *IEEE Trans. Autom. Control*, vol. 54, no. 6, pp. 1193–1203, Jun. 2009.
[29] A. Abate, M. Prandini, J. Lygeros, and S. Sastry, "Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems," *Automatica*, vol. 44, no. 11, pp. 2724–2734, Oct. 2008.
[30] X. D. Koutsoukos and D. Riley, "Computational methods for verification of stochastic hybrid systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Human*, vol. 38, no. 2, pp. 385–396, Mar. 2008.
[31] A. A. Julius and G. J. Pappas, "Probabilistic testing for stochastic hybrid systems," in *Proc. IEEE Conf. Decision Control*, Cancun, Mexico, Dec. 2008, pp. 4030–4035.
[32] A. Abate, J.-P. Katoen, J. Lygeros, and M. Prandini, "Approximate model checking of stochastic hybrid systems," *Eur. J. Control*, vol. 16, no. 6, pp. 624–641, 2010.
[33] M. Li and R. Kumar, "Reduction of automated test generation for Simulink/Stateflow to reachability and its novel resolution," in *Proc. 9th IEEE Int. Conf. Autom. Sci. Eng.*, Madison, WI, USA, Aug. 2013, pp. 1089–1094.
[34] A. Fehnker and F. Ivancic, "Benchmarks for hybrid systems verification," *Hybrid Syst.: Computation and Control*, vol. 2293 of LNCS, pp. 326–341, 2004.
[35] C. Baier and J.-P. Katoen, *Principles of Model Checking*. Cambridge, MA, USA: MIT, 2008.
[36] E. M. Clarke, O. Grumberg, and D. A. Peled, *Model Checking*. Cambridge, MA: MIT Press, 1999.
[37] R. Kumar and S. Takai, "Decentralized prognosis of failures in discrete event systems," *IEEE Trans. Autom. Control*, vol. 55, no. 1, pp. 48–59, Jan. 2010.
[38] J. Chen and R. Kumar, "Failure prognosability of stochastic discrete event systems," in *Proc. Amer. Control Conf.*, Portland, OR, USA, Jun. 2014, pp. 2041–2046.
[39] J. Chen and R. Kumar, "Stochastic failure prognosability of discrete event systems," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1570–1581, Jun. 2015.
[40] F. Mueller, "Challenges for cyber-physical systems: Security, timing analysis and soft error protection," in *Proc. Nat. Workshop on High Confidence Software Platforms for Cyber-Physical Syst.: Res. Needs and Roadmap*, Nov. 2006, pp. 1–3.



Jun Chen (S'11–M'14) received the B.S. degree from Zhejiang Uniersity, Hangzhou China, in 2009, and the Ph.D. degree from Iowa State University, Ames IA, USA, in 2014, both in electrical engineering.

He joined Idaho National Laboratory, Idaho Falls, ID, USA, in November 2014, where he is currently a Postdoctoral Research Associate, working on modeling, simulation and operations optimization of hybrid energy systems. His current research interests include discrete-event systems, cyber-physical systems, power and energy systems, and stochastic systems, together with their fault diagnosis and prognosis, resilient control, optimization and information security.

Dr. Chen was a recipient of the Exceptional Contributions Program Award from Idaho National Laboratory, the Research Excellence Award from Iowa State University, and a Provost Graduate Fellowship from University of Central Florida. Since 2013, he has been a TPC member for *Chinese Control and Decision Conference*.

Ratnesh Kumar (S'87–M'90–SM'00–F'07) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, in 1987, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Texas, Austin, TX, USA, in 1989 and 1991, respectively.

He has been a Professor of electrical and computer engineering with Iowa State University, Ames, IA, USA, since 2002. Prior to this, he held a faculty position with the University of Kentucky (1991–2002) in electrical and computer engineering and has held visiting positions with the University of Maryland, Applied Research Laboratory (at Penn State University), NASA Ames, Idaho National Laboratory, and United Technologies Research Center. His research interests include model-based design of embedded software, web-services, networks and cyberphysical systems, sensors and their networks with application to agriculture, power systems and energy harvesting. He is or has been an associate editor of *ACM Transactions on Embedded Computing Systems*, *SIAM Journal on Control and Optimization*, and *Journal of Discrete Event Dynamical Systems*.

Prof. Kumar was the recipient of Gold Medals for Best EE undergrad and Best All Rounder at IIT Kanpur, Best Dissertation Award at UT Austin. He is or has been an associate editor for the IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, the IEEE Control Systems Society, the IEEE Robotics and Automation Systems Society, and the IEEE Workshop on Software Cybernetics.