Safe Diagnosis of Stochastic Discrete Event Systems by Constructing Safe Verifier

Fuchun Liu and Pengbiao Yang

1 Introduction

Due to the practical and theoretical importance, failure diagnosis, which aims to timely identify and exactly characterize the occurrence of incipient faults, has received considerable attention in recent years (for example, [1–4] and the references, therein). In particular, Sampath et al. [3] proposed an approach for failure diagnosis of discrete event system (DES), in which a diagnoser was constructed to perform the on-line detection and off-line verification of the diagnosability properties of the system. Thorsley et al. [4] extended the framework of [3] to stochastic DESs. For the decentralized systems, Qiu and Kumar [2] and Liu et al. [5] investigated the failure diagnosis issue of the decentralized classical and stochastic DESs by constructing the local diagnosers.

Motivated by the fact that the complexity of constructing such diagnosers in above references is exponential in the cardinality of the states of the system, Yoo and Lafortune [6] presented a polynomial-time verification for the diagnosability of partially observed DESs. In [7, 8], two polynomial test methods for stochastic diagnosability of DESs were developed by constructing different diagnosis models. Moreira et al. [9] proposed a polynomial-time failure diagnosis approach for decentralized DESs. In our prior work [10], the diagnosability and safe diagnosability of fuzzy DESs were studied, respectively, in both of which the complexity of constructing the models for failure diagnosis are polynomial-time.

This paper aims to present a new approach to deal with the safe diagnosis problem for stochastic DESs by constructing the safe verifier, which was investigated in our previous work by constructing the safe diagnosers [11]. More specifically, after formalizing the notion of safe diagnosability of stochastic DESs,

F. Liu (✉) · P. Yang
School of Computers, Guangdong University of Technology,
Guangzhou 510006, China
e-mail: fliu2011@163.com

Z. Deng (ed.), *Proceedings of 2017 Chinese Intelligent Automation Conference*,
Lecture Notes in Electrical Engineering 458,
https://doi.org/10.1007/978-981-10-6445-6_57

we construct a non-fault automaton to represent the specification language. Then, a recognizer of the illegal language is introduced to distinguish the forbidden strings from the system behaviors, and a safe verifier is constructed to realize the safe diagnosis of the system. In particular, a necessary and sufficient condition for the verification of safe diagnosability of stochastic DESs is proposed. It is worth noting that the construction of the safe verifier is polynomial-time in the number of states and events of the system.

2 Notations and Preliminaries

A stochastic DES is modeled by a stochastic automaton $G = (X, \Sigma, \eta_G, x_0)$, where X is a set of states with the initial state x_0, Σ is the set of events, $\eta_G : X \times \Sigma \times X \to [0, 1]$ is a partial state transition probability function.

Define a partial transition function $\delta_G : X \times \Sigma \to X$ as: $\delta_G(x, \sigma) = x'$ if and only if $\eta_G(x, \sigma, x') > 0$, where $x, x' \in X$ and $\sigma \in \Sigma$. The event set Σ is partitioned into the observable event set Σ_o and the unobservable event set Σ_{uo}, $\Sigma_f \subseteq \Sigma_{uo}$ is the failure event set. Generally, Σ_f is partitioned into a set of failure types $\Sigma_f = \Sigma_{f1} \cup \Sigma_{f2} \ldots \cup \Sigma_{fm}$.

For convenience, we introduce the notations from [1, 3, 4, 11]: Let $s \in L$, denote $pre\,(s)$ and $suf\,(s)$ as the prefixes and suffixes of s, respectively. $L_\sigma(G, x) = \{u\sigma : \Pr(u\sigma|x) > 0, u \in \Sigma_{uo}^*, \sigma \in \Sigma_o\}$, and $L/s = \{t \in \Sigma^* : st \in L\}$ represents the post-language. $\Psi(\Sigma_{fi}) = \{s \in L : s_l \in \Sigma_{fi}\}$, where s_l is the final event of s. $P : \Sigma^* \to \Sigma_o^*$ is the projection defined in the usual manner.

As mentioned in [11], the purpose of safe diagnosis is to prevent local faults developing into failures that can lead to serious hazards. So we should prevent the system executing a forbidden string from Γ_i after a failure of type f_i, where $\{\Gamma_i \subseteq \Sigma^* : i = 1, 2, \ldots m\}$ is a class of forbidden string sets. The set of such illegal strings is called illegal language, denoted by $\Re_f^i$, which is defined as $\Re_f^i = \{\omega \in L/s : s \in \Psi(\Sigma_{fi}), \Gamma_i \in \omega\}$, where $\Gamma_i \in \omega$ represents that there is $t \in \Gamma_i$ such that t is a substring of ω.

Definition 1 [11]: A language L generated by a stochastic automaton $G = (X, \Sigma, \eta_G, x_0)$ is said to be safe diagnosable if for any $\varepsilon > 0$,

$$(\exists n_0 \in N)\,(\forall s \in \Psi(\Sigma_{fi}))\,(\forall t \in L/s \wedge ||t|| \geq n_0)\,(\exists v \in pre(t))\,(pre(v) \cap \Re_f^i = \varnothing)$$

and, at least one of the following conditions holds:

1. $\omega \in P^{-1}[P(sv)] \Rightarrow \Sigma_{fi} \in \omega$, i.e., $D(sv) = 1$,
2. $\Pr(t : D(sv) = 0|v \in pre(t)) < \varepsilon$.

where the diagnosability function $D : \Sigma^* \to \{0, 1\}$ is defined as follows:

$$D(sv) = \begin{cases} 1, & \text{if } \omega \in P^{-1}[P(sv)] \Rightarrow \Sigma_{fi} \in \omega, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

3 Construction of Safe Verifier of Stochastic DESs

Let $G = (X, \Sigma, \eta_G, x_0)$ be a stochastic DES, $\Re_f^i$ be the illegal language and Γ_i be the set of forbidden strings. In order to construct the recognizer G_r, we first define a label set of forbidden strings from Γ_i as $LB = \{N, F, B\}$, where the label N represents that the failure does not occur in the behavior of system, F represents that the system executes the failure but does not execute any forbidden string from Γ_i after the occurrence of that failure, and B means that the behavior of system containing a forbidden string from Γ_i after the occurrence of a failure.

Algorithm 1: Construction of safe verifier of stochastic DESs Step 1: Construct a non-fault automaton $H = (X', \Sigma_N, \eta_H, \delta_H, x_0')$ to generate the sublanguage of L without fault events, where $\Sigma_N = \Sigma - \Sigma_f$, X' is a set of finite states with the initial state $x_0' = x_0$, for any $x' \in X$, if there exists $\omega \in \Sigma^*$ satisfying $\Sigma_f \notin \omega$ and $\eta_G(x_0, \omega, x') > 0$, then $x' \in X'$, the state transition probability function $\eta_H : X' \times \Sigma \times X' \to [0, 1]$ is defined as $\eta_H(x, \sigma, x') = \eta_G(x, \sigma, x')$, for $\forall x, x' \in X'$, $\sigma \in \Sigma_N$ satisfying $\eta_G(x, \sigma, x') > 0$, the state transition function, $\delta_H : X' \times \Sigma \to X'$ is defined as $\delta_H(x, \sigma) = x'$ if and only if $\eta_H(x, \sigma, x') > 0$.

Step 2: Construct the recognizer of the illegal language $\Re_f^i$ as a stochastic automaton $G_r = (Q_r, \Sigma, \eta_r, \delta_r, p_0)$, where $Q_r \subseteq X \times LB$ is a set of finite states with the initial state $p_0 = (x_0, N)$, η_r and δ_r are defined in the following Definitions 2 and 3, respectively.

Step 3: Construct the safe verifier G_v based on the recognizer G_r, the $G_v = (Q_v, \Sigma_o, \delta_v, \eta_v, q_0)$ is a finite state automaton, where Q_v is a set of finite states with the initial state $q_0 = (x_0, N, y_0)$, $Q_v \subseteq X \times LB \times Y$, where $Y = X' \cup \{E\}$, E represents empty, η_v and δ_v are defined in the following Definitions 4 and 5, respectively.

Definition 2 The partial state transition probability function of recognizer G_r is defined as $\eta_r : Q_r \times \Sigma \times Q_r \to [0, 1]$: for $p_i = (x_i, lb_i)$, $p_j = (x_j, lb_j) \in Q_r$ and $\sigma \in \Sigma$, $\eta_r(p_i, \sigma, p_j) = \eta_G(x_i, \sigma, x_j)$.

Definition 3 The partial state transition function of recognizer G_r, is defined as $\delta_r : Q_r \times \Sigma \to Q_r$: for $p_i = (x_i, lb_i) \in Q_r$, $lb_i \in LB$ and $\sigma \in \Sigma$,

$$\delta_r(p_i, \sigma) = \begin{cases} (\delta_G(x_i, \sigma), N), & \text{if } \sigma \notin \Sigma_f, lb_i = N, \\ (\delta_G(x_i, \sigma), F), & \text{if } (\sigma \in \Sigma_f, lb_i = N) \vee lb_i = F, \\ (\delta_G(x_i, \sigma), B), & \text{if } (suf(s_2) \cap \Gamma_i \neq \varnothing, \text{where} \\ & s\sigma = s_1\sigma_f s_2, \sigma_f \in \Sigma_f) \vee lb_i = B. \end{cases} \quad (2)$$

Definition 4 The state transition probability function $\eta_v : Q_v \times \Sigma_o \times Q_v \to [0,1]^2$ of safe verifier G_v is defined as: for any $q_1 = (x_1, lb_1, y_1)$, $q_2 = (x_2, lb_2, y_2) \in Q_v$, $\sigma \in \Sigma_o$, $\eta_v(q_1, \sigma, q_2) = (\theta, \theta') > 0$ holds, where

$$\theta' = \sum_{s \in L_\sigma(H, y_1)} \eta_H(y_1, s, y_2), \quad \theta = \sum_{s \in L_\sigma(G, x_1)} \eta_G(x_1, s, x_2).$$

Definition 5 The state transition function of $\delta_v : Q_v \times \Sigma_o \to Q_v$ is defined as:

$$\delta_v(q_1, \sigma) = \begin{cases} (\delta_r(p_1, s), E), & \text{if } \delta_H(y_1, \sigma) \text{ is undefined or } y_1 = E, \\ (\delta_r(p_1, s), \delta_H(y_1, \sigma)), & \text{otherwise,} \end{cases} \tag{3}$$

where for any $q_1 = (x_1, lb_1, y_1) \in Q_v$, $\sigma \in \Sigma_o$, $s \in L_\sigma(G, x_1)$, $p_1 = (x_1, lb_1) \in Q_r$, state E means that there is no transition originate from state y_1 with the event of σ.

Remark It is worth noting that the construction of the safe verifier is polynomial-time in the number of states and events of the system.

4 Necessary and Sufficient Condition of Safe Diagnosability

Before proposing the necessary and sufficient condition, we introduce some concepts about the safe verifier.

Definition 6 For state $q_i = (x_i, lb_i, y_i) \in Q_v$, if $lb_i = N$, then q_i is called a normal state. The set of all normal states is denoted as Q_v^N, if $lb_i = B$, then q_i is called a B state, and the set of all B states is denoted as Q_v^B, if $lb_i = F \wedge y_i \neq E$, then q_i is called a fault state, and the set of all fault states is denoted as Q_v^F, where $Q_v^N, Q_v^B, Q_v^F \subseteq Q_v$.

Definition 7 If there exists the sequence of states $q_k, q_{k+1}, q_{k+2}, \ldots q_l \in Q_v$ and events $\sigma_k, \sigma_{k+1}, \sigma_{k+2}, \ldots \sigma_l \in \Sigma_o$, $0 \leq k \leq l$ such that

$$q_k \xrightarrow{\sigma_k, (\theta_k, \theta'_k)} q_{k+1} \xrightarrow{\sigma_{k+1}, (\theta_{k+1}, \theta'_{k+1})} q_{k+2} \ldots q_l \xrightarrow{\sigma_l, (\theta_l, \theta'_l)} q_k, \tag{4}$$

then the sequence of states $q_k, q_{k+1}, q_{k+2}, \ldots q_l \in Q_v$ forms a state cycle, denoted as $cl = (q_k, q_{k+1}, \ldots q_l)$, and the set of the states in the cycle cl is denoted as cl'.

Definition 8 Assume that $cl = (q_k, q_{k+1}, q_{k+2}, \ldots q_l)$ is a state cycle of safe verifier G_v, and its probability of transition is $(\vartheta, \vartheta') = (\theta_k \theta_{k+1} \ldots \theta_l, \theta'_k \theta'_{k+1} \ldots \theta'_l)$, if $q_i \in Q_v^B$, $(k \leq i \leq l)$ and $\vartheta = 1$, then the state cycle cl is called a recurrent B state cycle, denoted as $cl^B_{recurrent}$.

Theorem 1 *Let L be the generated language of stochastic DES* $G = (X, \Sigma, \eta_G, x_0)$ *and* $G_v = (Q_v, \Sigma_o, \delta_v, \eta_v, q_0)$ *be the safe verifier. L is safe diagnosable, if and only if,* G_v *satisfies the following condition: There does not exist the states* $q_1 \in Q_v^F \vee Q_v^N$, $q_2 \in Q_v^B$, $q_3 \in cl^{B\prime}_{recurrent}$ *such that* $\delta_v(q_1, \sigma) = q_2$ *and* $\delta_v(q_2, \alpha) = q_3$, where $\sigma \in \Sigma_o$, $\alpha \in \Sigma^*$.

Due to the limitation of space, the proof is omitted. Next, some examples are provided to illustrate Algorithm 1 and Theorem 1.

Example 1 For comparison with the method proposed in [11], we consider the same stochastic system $G_1 = (X, \Sigma, \eta_G, x_0)$ as that in [11] (Example 3 in [11]), which is shown in Fig. 1, where $\Sigma_0 = \{a, b\}$, $\Sigma_f^i = \{\sigma_f\}$ and $\Gamma_i = \{b\}$. In [11], it has been proved that G_1 is not safe diagnosable by constructing the safe diagnoser.

In the following, we verify this result by constructing the safe verifier proposed in this paper (i.e., by using Algorithm 1 and Theorem 1).

According to Algorithm 1, we construct the non-fault automaton H_1 shown in Fig. 2 by Step 1 and the recognizer G_{r1} shown in Fig. 3 by Step 2.

Then we construct the safe verifier G_{v1} shown in Fig. 4 by Step 3.

Note that in Fig. 4, $\delta_v((0, N, 0), b) = (3, B, E)$,$\delta_v((3, B, E), b) = (3, B, E)$ and $\delta_v((2, F, 1), b) = (3, B, E)$, $\delta_v((3, B, E), b) = (3, B, E)$ where $(3, B, E) \in cl^{B\prime}_{recurrent}$. By Theorem 1, G_1 is not safe diagnosable, which coincides with the result in [11].

Example 2 Consider the stochastic system $G_2 = (X, \Sigma, \eta_G, x_0)$ shown in Fig. 5, which is the same example as that in [11] (Example 4 in [11]), where

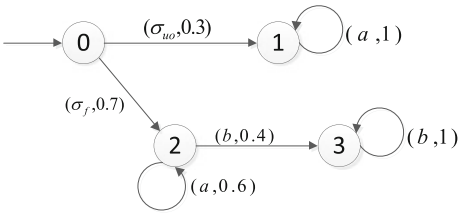

Fig. 1 Stochastic system G_1

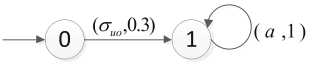

Fig. 2 Non-fault automaton H_1

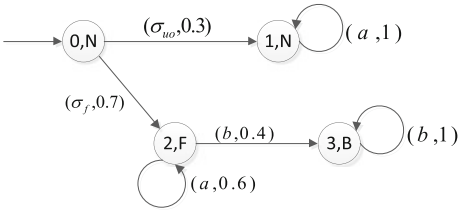

Fig. 3 Recognizer G_{r1}

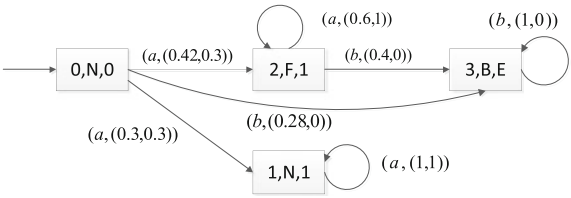

Fig. 4 Safe verifier G_{v1}

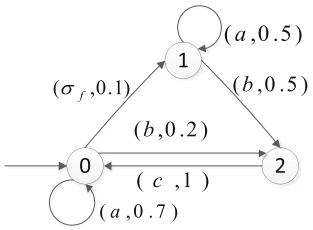

Fig. 5 System G_2

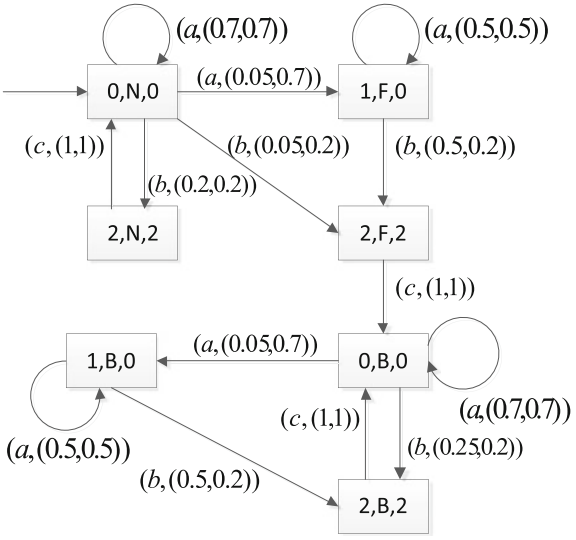

Fig. 6 Safe verifier G_{v2}

$\Sigma_0 = \{a, b, c\}$, $\Sigma_f^i = \{\sigma_f\}$ and $\Gamma_i = \{c\}$. In [11], it is proved that G_2 is safe diagnosable. Next, we verify the result by using the methods proposed in this paper.

Similar to Example 1, the safe verifier G_{v2} is constructed as that shown in Fig. 6. Note that G_{v2} satisfies the conditions of Theorem 1. Therefore, we conclude G_2 being safe diagnosable, which coincides with the result obtained in [11].

5 Conclusion

Motivated by the fact that the complexity of constructing safe diagnosers to deal with safe diagnosability of stochastic DESs in [11] is exponential, a new algorithm is proposed by constructing the safe verifier to realize safe diagnosis of stochastic DESs, which is polynomial-time in the number of states and events of the system.

Acknowledgements This work is supported by the National Natural Science Foundation (61673122, 61273118), the Public Welfare Research and Capacity Building Project of Guangdong Province (2015A030402006), the Provincial Major Program of Guangdong Province (2014KZDXM033), the Major Award Training Program of School of Computers of GDUT (2016PY01) of China.

References

1. Paoli A, Lafortune S (2005) Safe diagnosability for fault-tolerant supervision of discrete-event systems. IEEE Trans Autom Control 41(8):1335–1347
2. Qiu W, Kumar R (2006) Decentralized failure diagnosis of discrete event systems. IEEE Transaction on Systems, Man, and Cybernetics-part A: Systems and Humans 36(2):384–395
3. Sampath M, Sengupta R, Lafortune S, Sinnamohideen K, Teneketzis D (1995) Diagnosability of discrete-event systems. IEEE Trans Autom Control 40(9):1555–1575
4. Thorsley D, Teneketzis D (2005) Diagnosability of stochastic discrete-event systems. IEEE Trans Autom Control 50(4):476–492
5. Liu F, Qiu D, Xing H, Fan Z (2008) Decentralized diagnosis of stochastic discrete event systems. IEEE Trans Autom Control 53(2):535–546
6. Yoo T, Lafortune S (2002) Polynomial-time verification of diagnosability of partially observed discrete-event systems. IEEE Trans Autom Control 47(9):1491–1495
7. Chen J, Kumar R (2012) Polynomial test for stochastic diagnosability of discrete-event systems. IEEE Trans Autom Sci Eng 10(4):969–979
8. Luo M, Sun F, Li Y (2011) A polynomial algorithm for testing diagnosability of stochastic discrete event systems. In: 8th Asian Control Conference (ASCC). pp 1048–1053
9. Moreira MV, Jesus TC, Basilio JC (2011) Polynomial time verification of decentralized diagnosability of discrete event systems. IEEE Trans Autom Control 56(7):1679–1684
10. Liu F (2015) Safe Diagnosability of fuzzy discrete-event systems and a polynomial-time verification. IEEE Trans Fuzzy Syst 23(5):1534–1544
11. Liu F, Qiu D (2008) Safe diagnosability of stochastic discrete event systems. IEEE Trans Autom Control 53(5):1291–1296