
Discrete Event Dynamic Systems: Theory and Applications, 14, 171–202, 2004

Diagnosis of Intermittent Faults

OLIVIER CONTANT olivier@eecs.umich.edu
Department of Electrical Engineering and Computer Science, The University of Michigan, 1301 Beal Avenue, Ann Arbor, MI 48109-2122 USA

STÉPHANE LAFORTUNE stephane@eecs.umich.edu
Department of Electrical Engineering and Computer Science, The University of Michigan, 1301 Beal Avenue, Ann Arbor, MI 48109-2122 USA

DEMOSTHENIS TENEKETZIS teneket@eecs.umich.edu
Department of Electrical Engineering and Computer Science, The University of Michigan, 1301 Beal Avenue, Ann Arbor, MI 48109-2122 USA

Abstract. The diagnosis of ''intermittent'' faults in dynamic systems modeled as discrete event systems is considered. In many systems, faulty behavior often occurs intermittently, with fault events followed by corresponding ''reset'' events for these faults, followed by new occurrences of fault events, and so forth. Since these events are usually unobservable, it is necessary to develop diagnostic methodologies for intermittent faults. Prior methodologies for detection and isolation of permanent faults are no longer adequate in the context of intermittent faults, since they do not account explicitly for the dynamic behavior of these faults. This paper addresses this issue by: (i) proposing a modeling methodology for discrete event systems with intermittent faults; (ii) introducing new notions of diagnosability associated with fault and reset events; and (iii) developing necessary and sufficient conditions, in terms of the system model and the set of observable events, for these notions of diagnosability. The definitions of diagnosability are complementary and capture desired objectives regarding the detection and identification of faults, resets, and the current system status (namely, is the fault present or absent). The associated necessary and sufficient conditions are based upon the technique of ''diagnosers'' introduced in earlier work, albeit the structure of the diagnosers needs to be enhanced to capture the dynamic nature of faults in the system model. The diagnosability conditions are verifiable in polynomial time in the number of states of the diagnosers.

Keywords: diagnosability, intermittent faults, fault diagnosis, fault detection

1. Introduction

Practical experience has shown that detection and isolation of many classes of faults in dynamic systems can be approached as a problem of state estimation and inferencing for discrete event systems (Aghasaryan et al., 1998; Benveniste, 2003; Bouloutas, 1990; Console, 2000; Debouk et al., 2000; Garcia et al., 2002; Jiang and Kumar, 2002; Jiang et al., 2002; Lafortune et al., 2001; Lamperti and Zanella, 1999; Lin, 1994; Lin et al., 1993; Lunze, 2000; Pandalai and Holloway, 2000; Pencolé, 2000; Pencolé et al., 2001; Sampath, 2001; Sampath et al., 1998, 1995, 1996; Sengupta, 2001; Sinnamohideen, 2001; Westerman et al., 1998; Hastrudi Zad et al., 1998). In many systems, faulty behavior often occurs intermittently, with fault events followed by corresponding ''reset'' events for these faults, followed by new occurrences of fault events, and so forth. In hardware systems, intermittent

faults are typically caused by bad electrical contacts (e.g., faulty relays), ''sticky'' components (e.g., stuck valves), overheating of chips, noisy measurements from sensors, power surges, and so forth. Intermittent faults occur in software systems as well; consider for instance exceptions and interrupts that are caused by some unknown ''bugs'' and that lead to crashes and reboots. The methodologies used in Aghasaryan et al. (1998), Benveniste et al. (2003), Bouloutas (1990), Console (2000), Debouk et al. (2000), Garcia (2002), Jiang and Kumar (2002), Lafortune et al. (2001), Lamperti and Zanella (1999), Lin (1994), Lin et al. (1993), Lunze (2000), Pandalai and Holloway (2000), Pencolé (2000), Pencolé et al. (2001), Sampath (2001), Sampath et al. (1998, 1995, 1996), Sengupta (2001), Sinnamohideen (2001), Westerman et al. (1998) and Hastrudi Zad et al. (1998) assume that once faults occur, they remain in effect permanently; hence, the terminology ''failures'' is often used for these permanent faults. Furthermore, to the best of our knowledge, diagnostic methodologies developed in the field of model-based reasoning in artificial intelligence (which are close in spirit to the discrete event systems methodologies, since they are also based on qualitative system models) are also geared towards the diagnosis of permanent faults; see, for example, Darwiche and Provan (1996), Dvorak and Kuipers (1992), Provan and Chen (1998, 1999), and Williams and Nayak (1996).

Methodologies for diagnosing permanent faults are no longer adequate in the context of intermittent faults, since they do not account explicitly for the dynamic behavior of these faults that manifests itself in the form of alternating (unobservable) fault and reset events. The work in Aghasaryan et al. (1998) and Benveniste et al. (2003) allows intermittent faults but does not propose a systematic framework for their detection and isolation. The linear temporal logic approach to failure diagnosis presented in Jiang and Kumar (2002) may prove a viable approach for detection and isolation of intermittent faults, but this remains to be explored. The recent work (Jiang et al., 2002) presents a state-based modeling of faults (and implicitly their resets) and focuses on the diagnosis of the number of occurrences of faults. Our focus in this paper is different from that in Jiang et al. (2002). Our main concern is the diagnosis of the current status of the system (i.e., which faults are present, which faults have never occurred, and which faults are reset). Our principal contributions in this regard are:

- A novel modeling methodology for discrete event systems with intermittent faults and their associated reset events.

- A set of four new definitions of diagnosability associated with the fault and reset events.

- Necessary and sufficient conditions, in terms of the system model and the set of observable events, for each of the four notions of diagnosability.

We consider untimed models of discrete event systems. The four definitions of diagnosability are complementary and capture a hierarchy of desired objectives regarding the detection and identification of faults, resets, and the current system status (namely, which faults are present, absent, or reset). The associated necessary and sufficient conditions are based upon the techniques introduced in the diagnoser approach of

Sampath et al. (1995), albeit the structure of the diagnoser automata needs to be altered to capture the dynamic nature of faults in the system model. The enhancements to the ''diagnoser approach'' are due to the introduction of new labels for faults and resets in the construction of diagnosers, which in turn leads to the consideration of new types of indeterminate cycles (Sampath et al., 1995) that serve to characterize violations of the four types of diagnosability. The necessary and sufficient conditions are verifiable in polynomial time in the number of states of the diagnosers.

This paper is organized as follows. Section 2 first presents some necessary background and then focuses on the modeling of intermittent faults and on how their dynamic behavior is captured by three types of labels. Section 3 presents the four notions of diagnosability that are proposed to thoroughly capture desired objectives in the context of intermittent faults. Modified diagnosers that explicitly account for the new label types in building state estimates are described in Section 4. Sections 5 and 6 then develop the four sets of necessary and sufficient conditions associated with the four notions of diagnosability. A detailed example of the modeling of intermittent faults and the ensuing system analysis is given in Section 7. Conclusions appear in Section 8.

2. Modeling of System and Intermittent Faults

2.1. System Model and Assumptions

We assume that the reader is familiar with basic notions in finite-state automata and regular languages (see, for example, Cassandras and Lafortune, 1999). The system to be diagnosed is modeled as an automaton

$$G = (X, \Sigma, \delta, x_0) \tag{1}$$

where X is the state space, Σ is the set of events, δ is the partial transition function, and x_0 is the initial state of the system. Model G accounts for the normal and failed behavior of the system. The behavior of the system is described by the prefix-closed language $L(G)$ generated by G. Henceforth, we shall denote $L(G)$ by L. L is a subset of Σ^*, where Σ^* denotes the Kleene closure of the set Σ. The language L generated by G is assumed to be live. This means that there is a transition defined at each state x in X, i.e., the system cannot reach a point at which no event is possible. The liveness assumption on L is made for the sake of simplicity. With slight modifications, all the main results of this paper hold true when the liveness assumption is relaxed (cf. the results in Sampath et al., 1998).

Some of the events in Σ are observable, i.e., their occurrence can be observed, while the rest are unobservable. Thus, the event set Σ is partitioned as $\Sigma = \Sigma_o \dot{\cup} \Sigma_{uo}$, where Σ_o represents the set of observable events and Σ_{uo} represents the set of unobservable events. The observable events in the system may be one of the following: commands issued by the controller, sensor readings immediately after the execution of the above commands, and changes of sensor readings. The unobservable events may be fault events, reset events, or other events that cause changes in the system state not recorded by sensors. See Sampath et al. (1996) for a methodology on how to construct the system model G from models of

system components and sensor readings. We assume that there does not exist in G any cycle of unobservable events, i.e., $\exists\, n_0 \in \mathbb{N}$ such that $\forall\, s_1 t s_2 \in L, t \in \Sigma_{uo}^* \Rightarrow \| t \| \leq n_o$ where $\| t \|$ is the length of trace t. This assumption ensures that observations occur with some regularity. Since detection of faults is based on observable transitions of the system, we require that G does not generate arbitrarily long sequences of unobservable events.

In the context of diagnosis of intermittent faults, let $\Sigma_f \subseteq \Sigma$ denote the set of fault events and let $\Sigma_r \subseteq \Sigma$ denote the corresponding set of fault reset events which should be diagnosed. In this regard, the set Σ_f is assumed to be composed of m different fault events, $\Sigma_f = \{f_1, \ldots, f_m\}$, and the set Σ_r is assumed to be composed of the corresponding resets, $\Sigma_r = \{r_1, \ldots, r_m\}$. Each fault event f_i has its corresponding fault reset event r_i, where r_i cannot happen until f_i occurs at least once. This last assumption points out the fact that we are dealing with intermittent faults hence each fault event can potentially be reset; if known permanent faults are present, they can be handled by the methodology in Sampath et al. (1995). Without loss of generality, we assume that $\Sigma_f \subseteq \Sigma_{uo}$ and $\Sigma_r \subseteq \Sigma_{uo}$, since an observable fault event or an observable fault reset event can be diagnosed trivially. The objective is to identify the occurrence, if any, of the fault events and their corresponding reset events, while tracking the observable events generated by the system.

The assumptions made above on the system under investigation are required in all the results of this paper (even if not explicitly stated).

To define diagnosability, we need the following notation. The empty trace is denoted by ε. Let $\bar{s}$ denote the prefix-closure of trace s where $s \in \Sigma^*$. We denote by L/s the postlanguage of L after s, i.e.,

$$\frac{L}{s} := \{t \in \Sigma^* : st \in L\}$$

We define the projection $P: \Sigma^* \to \Sigma_o^*$ in the usual manner (Ramadge and Wonham, 1989)

$$\begin{aligned} P(\varepsilon) &= \varepsilon \\ P(\sigma) &= \sigma \qquad \text{if } \sigma \in \Sigma_o \\ P(\sigma) &= \varepsilon \qquad \text{if } \sigma \in \Sigma_{uo} \\ P(s\sigma) &= P(s)P(\sigma) \quad \text{where } s \in \Sigma^*, \sigma \in \Sigma \end{aligned} \tag{2}$$

Thus, P simply ''erases'' the unobservable events in a trace. The inverse projection operator P_L^{-1} is defined as

$$P_L^{-1}(y) = \{s \in L : P(s) = y\} \tag{3}$$

We will write $\Psi(f_i)$ to denote the set of all traces of L that end with the fault event f_i. That is,

$$\Psi(f_i) := \{sf_i \in L\} \tag{4}$$

Similarly, we will write $\Psi(r_i)$ to denote the set of all traces of L that end with the reset event r_i. That is,

$$\Psi(r_i) := \{sr_i \in L\} \tag{5}$$

Consider $\sigma \in \Sigma$ and $s \in \Sigma^*$. We use the notation $\sigma \in s$ to denote the fact that σ is an event in the trace s.

We define

$$X_o = \{x_o\} \cup \{x \in X : x \text{ has an observable event into it}\} \tag{6}$$

Let $L(G,x)$ denote the set of all traces that originate from state x of G. $L_o(G,x)$ denotes the set of all traces that originate from state x and end at the first observable event. $L_\sigma(G,x)$ denotes those traces in $L_o(G,x)$ that end with the particular observable event σ; s_f denotes the final event of trace s. Formally,

$$L_o(G,x) = \{s \in L(G,x) : s = u\sigma, u \in \Sigma_{uo}^*, \sigma \in \Sigma_o\} \tag{7}$$

and

$$L_\sigma(G,x) = \{s \in L_o(G,x) : s_f = \sigma\} \tag{8}$$

Finally, we define the non-deterministic automaton

$$G' = (X_o, \Sigma_o, \delta_{G'}, x_0) \tag{9}$$

the generator of the language

$$L(G') = P(L) = \{t : t = P(s) \quad \text{for some } s \in L\} \tag{10}$$

The elements X_o, Σ_o, and x_0 are as defined above. The transition relation of G' is given by $\delta_{G'} \subseteq (X_o \times \Sigma \times X_o)$ and is defined as follows:

$$(x, \sigma, x') \in \delta_{G'} \quad \text{if} \quad \delta(x,s) = x' \quad \text{for some } s \in L_\sigma(G,x) \tag{11}$$

2.2. *Modeling of Intermittent Faults*

As in Sampath et al. (1995), we use the notion of label to identify special changes in the status of the system. The labels are symbols that allow us to keep track of the occurrence of selected events along the system's evolution. We define the set of non-intermittent and present fault labels $\Delta_F = \{F_1, F_2, \ldots, F_m\}$, the set of reset fault labels $\Delta_{F^{IR}} = \{F_1^{IR}, F_2^{IR}, \ldots, F_m^{IR}\}$, the set of intermittent and present fault labels $\Delta_{F^{IP}} = \{F_1^{IP}, F_2^{IP}, \ldots, F_m^{IP}\}$, and the set $\Delta = \{N\} \cup \Delta_F \cup \Delta_{F^{IR}} \cup \Delta_{F^{IP}}$.

Next, we define the label function ℓ^R that will be applied to traces and subtraces in $L(G)$.

DEFINITION 1 *The label function* $\ell^R : \Sigma^* \to 2^\Delta$ *is defined as follows. Let* ω *be a trace in* Σ^*. *Then*

$$
\begin{aligned}
&\ell^R(\omega) = \{N\} && \text{if } \forall i : (f_i \notin \omega) \wedge (r_i \notin \omega),\\
&\text{and } \forall i \in \{1, \ldots, m\},\\
&F_i \in \ell^R(\omega) && \text{if } (f_i \in \omega) \wedge (r_i \notin \omega),\\
&F_i^{IR} \in \ell^R(\omega) && \text{if } \exists\ s, s' : (\omega = ss') \wedge [s \in \Psi(r_i)] \wedge (f_i \notin s'),\ \text{and}\\
&F_i^{IP} \in \ell^R(\omega) && \text{if } \exists\ s, s' : (\omega = ss') \wedge [s \in \Psi(f_i)] \wedge (r_i \in s) \wedge (r_i \notin s')
\end{aligned}
$$

Hence, if $\ell^R(\omega)$ is $\{N\}$, i.e., ''normal'', then no event from the set of fault events Σ_f and no event from the set of reset events Σ_r have occurred along the trace. If $\ell^R(\omega)$ contains the label F_i, then the fault event f_i has occurred along ω but the reset event r_i has not occurred along ω. If $\ell^R(\omega)$ contains the label F_i^{IR}, then both the fault event f_i and the reset event r_i have occurred at least one time or possibly multiple times along ω, but the last of the two to have occurred in ω is r_i. Finally, if $\ell^R(\omega)$ contains the label F_i^{IP}, then both the fault event f_i and the reset event r_i have occurred at least one time or possibly multiple times along ω, but the last of the two to have occurred in ω is f_i.

Figures 1 and 2 illustrate the label function ℓ^R. Figure 1 shows how the label evolves as a trace gets extended by the occurrence and re-occurrence of fault and reset events. In Figure 2, the notation $(x, \{F_1, F_2^{IR}, F_3^{IP}\})$ means that along a trace that leads to state x the events f_1, f_2, r_2, f_3, r_3 have occurred, the event r_2 was the last one to occur among f_2 and r_2, and the event f_3 was the last one to occur among f_3 and r_3.

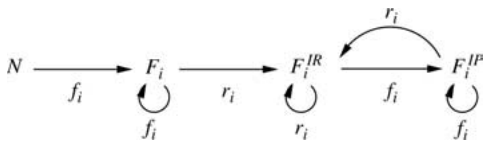


Figure 1. Label function evolution diagram.

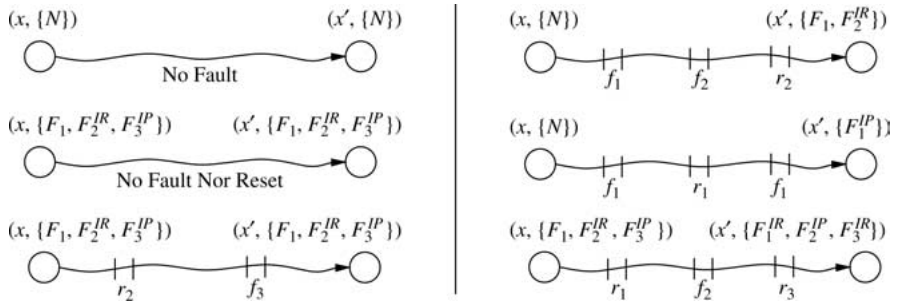


Figure 2. States and label function.

Remark 1: Let $\omega \in L(G)$.

i. If $\ell^R(\omega) = \{N\}$, then $F_i, F_i^{IR}, F_i^{IP} \notin \ell^R(\omega)$ for all i.

ii. If $F_i \in \ell^R(\omega)$ for some i, then $N \notin \ell^R(\omega), F_i^{IR} \notin \ell^R(\omega)$, and $F_i^{IP} \notin \ell^R(\omega)$.

iii. If $F_i^{IR} \in \ell^R(\omega)$ for some i, then $N \notin \ell^R(\omega), F_i \notin \ell^R(\omega)$, and $F_i^{IP} \notin \ell^R(\omega)$.

iv. If $F_i^{IP} \in \ell^R(\omega)$ for some i, then $N \notin \ell^R(\omega), F_i \notin \ell^R(\omega)$, and $F_i^{IR} \notin \ell^R(\omega)$.

2.3. *Recurrent Faults*

We define and motivate the notions of Σ_f-recurrent and Σ_r-recurrent languages.

DEFINITION 2 *Σ_f-recurrence and Σ_r-recurrence*

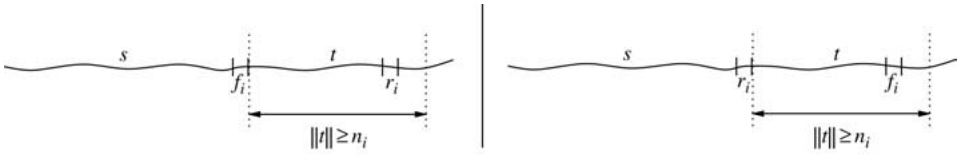
a. *A prefix-closed and live language L is said to be Σ_f-recurrent with respect to the set of fault events Σ_f and the set of reset events Σ_r if the following holds:*

$$\forall f_i \in \Sigma_f, i \in \{1, \ldots, m\}, \exists n_i \in \mathbb{N} \text{ such that}$$
$$[\forall s \in \Psi(f_i)] \left(\forall t \in \frac{L}{s} \right) [\| t \| \geq n_i \Rightarrow r_i \in t] \tag{12}$$

b. *A prefix-closed and live language L is said to be Σ_r-recurrent with respect to the set of fault events Σ_f and the set of reset events Σ_r if the following holds:*

$$\forall r_i \in \Sigma_r, i \in \{1, \ldots, m\}, \exists n_i \in \mathbb{N} \text{ such that}$$
$$[\forall s \in \Psi(r_i)] \left(\forall t \in \frac{L}{s} \right) [\| t \| \geq n_i \Rightarrow f_i \in t] \tag{13}$$

The above notions are motivated by the following considerations. We are concerned with the dynamic behavior of discrete event systems where failure and reset events occur continuously along any path of the systems's evolution; such a behavior is ensured by Σ_f-recurrence and Σ_r-recurrence. These notions imply that fault and reset events occur with



(a) Σ_f-recurrence (b) Σ_r-recurrence

Figure 3. Σ_f-recurrence and Σ_r-recurrence.

some regularity along any possible behavior of the system; as will be seen in Section 5, such regularity allows the diagnosis of these events. That is, Σ_f-recurrence (respectively, Σ_r-recurrence), along with some other conditions (identified in Section 5), ensure that there are instances where we are sure that certain faults are present in (respectively, absent from) the system.

3. Notions of Diagnosability

Intermittent faults are dynamic, that is, they can repeatedly occur and reset. This feature renders their diagnosis considerably more complicated and intricate than the diagnosis of permanent failures. The notion of diagnosability proposed in Sampath et al. (1995) for permanent failures relies on the fact that the status of faults remains fixed after their occurrence. In systems with intermittent faults, the fault's status evolves along with the system's evolution. Consequently, the notion of diagnosability proposed in Sampath et al. (1995) does not capture all the key issues associated with the diagnosis of intermittent faults. Since intermittent faults are dynamic, one can imagine several different notions of diagnosability, each providing a different amount of information about the status of faults. Ideally, one would want to detect the existence of instances where the status of faults (present or reset) is precisely known. A weaker notion of diagnosability would require that one may want to ensure detection of the occurrence of a fault, or detection of a fault's reset, without necessarily identifying any instance where the status of the fault is precisely known. These considerations motivate the definition of four types of diagnosability. Two of these types are related to the occurrence of intermittent faults; their ''dual'' notions are related to the reset of intermittent faults.

The first two notions of diagnosability, Type-P and Type-R, assert the presence of a fault or the absence of an intermittent fault at a specific instance.

DEFINITION 3 *Type-P diagnosability*
A prefix-closed and live language L is said to be Type-P diagnosable with respect to projection P, the set of fault events Σ_f, and the set of reset events Σ_r if the following holds:

$$[\forall i \in \{0, \ldots, m\}](\exists n_i \in \mathbb{N})[\forall s \in \Psi(f_i)]\left(\forall t \in \frac{L}{s}\right)[\| t \| \geq n_i \Rightarrow D_P]$$

where the diagnosability condition D_P is

$$\exists t' \leq t\colon w \in [P_L^{-1} P(st')] \Rightarrow [F_i \in \ell^R(w)] \vee [F_i^{IP} \in \ell^R(w)]$$

Type-P diagnosability, where P stands for present, implies that after a fault occurs along the system's evolution it is possible to identify an instance where, based on the available information, we are certain that the fault is present in the system. Such an instance is depicted in Figure 4. We use the label F_i^P to denote either F_i or F_i^{IP}.

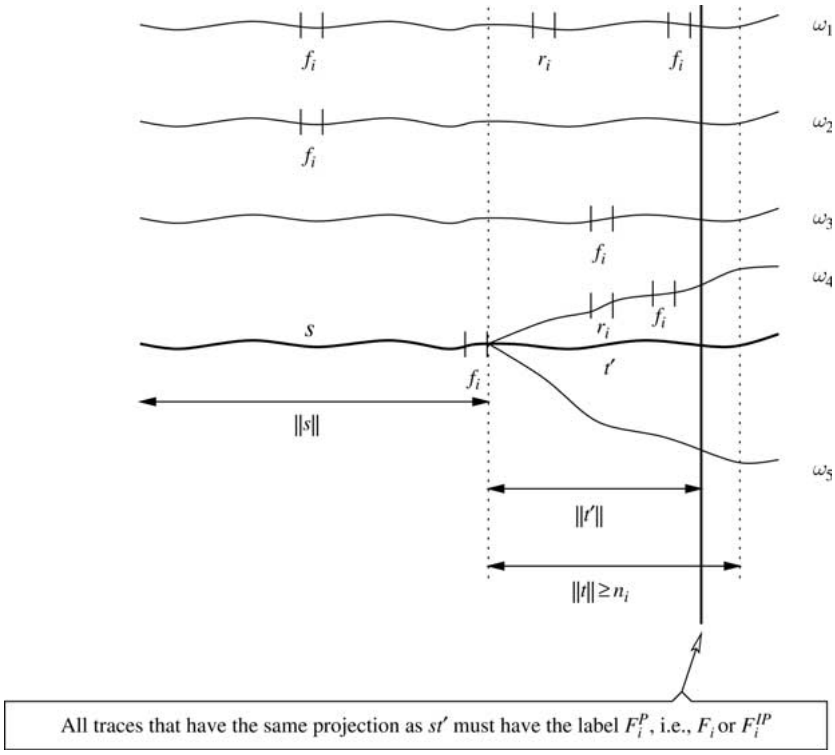


Figure 4. Type-P diagnosability.

DEFINITION 4 *Type-R diagnosability*
A prefix-closed and live language L is said to be Type-R diagnosable with respect to projection P, the set of fault events Σ_f*, and the set of reset events* Σ_r *if the following holds:*

$$[\forall i \in \{0, \ldots, m\}](\exists n_i \in \mathbb{N})[\forall s \in \Psi(r_i)]\left(\forall t \in \frac{L}{s}\right)[\| t \| \geq n_i \Rightarrow D_R]$$

where the diagnosability condition D_R *is*

$$\exists t' \leq t\colon w \in [P_L^{-1} P(st')] \Rightarrow [F_i^{IR} \in \ell^R(w)]$$

Type-R diagnosability, where R is to be interpreted as meaning reset, implies that after a fault reset occurs along the system's evolution, it is possible to identify an instance where we are certain that the fault is absent in the system. Such an instance is illustrated in Figure 5.

The following notions of diagnosability are weaker than Type-P and Type-R, as they allow the detection of the occurrence of a fault or the detection of a fault's reset without

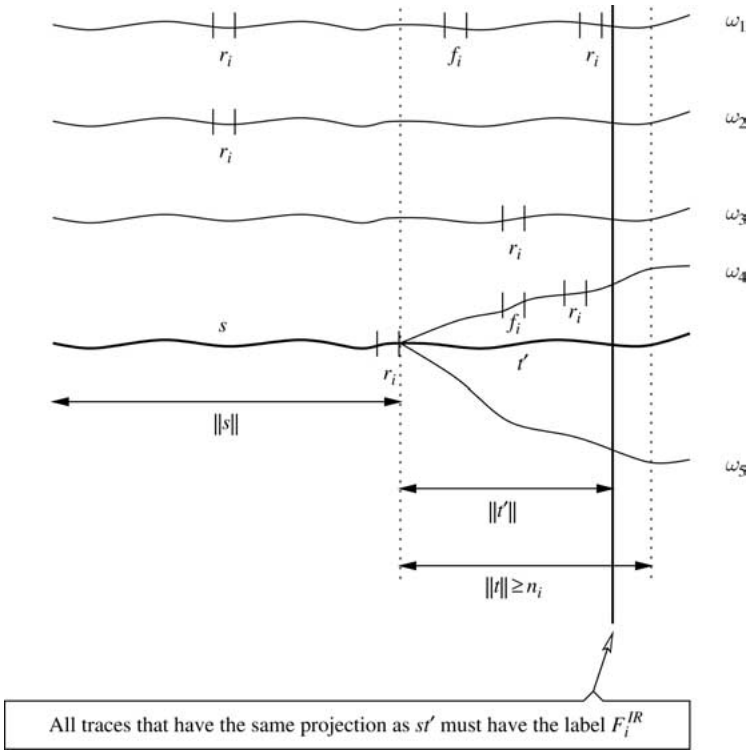


Figure 5. Type-R diagnosability.

necessarily identifying any instances where the status of the fault (present or reset) is precisely known.

DEFINITION 5 *Type-O diagnosability*
A prefix-closed and live language L is said to be Type-O diagnosable with respect to projection P, the set of fault events Σ_f*, and the set of reset events* Σ_r *if the following holds:*

$$[\forall i \in \{0, \ldots, m\}](\exists n_i \in \mathbb{N})[\forall s \in \Psi(f_i)]\left(\forall t \in \frac{L}{s}\right)[\| t \| \geq n_i \Rightarrow D_O]$$

where the diagnosability condition D_O *is*

$$\omega \in [P_L^{-1} P(st)] \Rightarrow [F_i \in \ell^R(\omega)] \vee [F_i^{IR} \in \ell^R(\omega)] \vee [F_i^{IP} \in \ell^R(\omega)]$$

Type-O diagnosability, where O stands for occurrence, has the following meaning. Suppose that fault f_i occurs along the system's evolution. Then, after at most n_i events, it is possible to identify the occurrence of f_i. This means that all possible system behaviors (i.e., all possible sequences of events in the system) which are compatible with

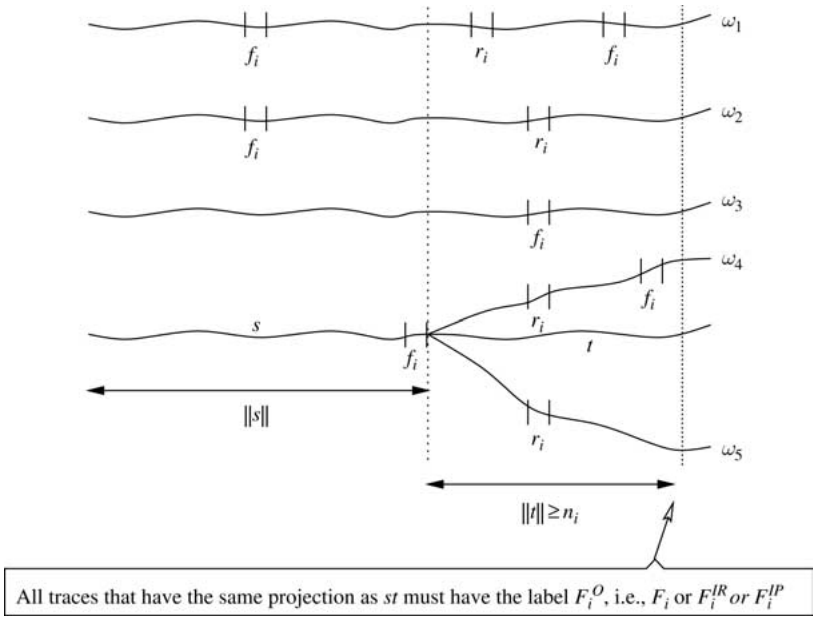


Figure 6. Type-O diagnosability.

the information available after n_i further events contain the fault event f_i. However, Type-O diagnosability does not guarantee perfect knowledge of the status of the fault event f_i at any stage. This means that along some of the possible traces f_i may be reset, along some others f_i may reset and reoccur more than once, yet along some others f_i may never be reset. This fact is depicted in Figure 6. For the sake of convenience, the label F_i^O shall be used to denote either F_i, F_i^{IR}, or F_i^{IP}.

Type-I diagnosability, where I is to be interpreted as meaning intermittent, is in some way the dual notion of Type-O diagnosability.

DEFINITION 6 *Type-I diagnosability*
A prefix-closed and live language L is said to be Type-I diagnosable with respect to projection P, the set of fault events Σ_f, and the set of reset events Σ_r if the following holds:

$$[\forall i \in \{0, \ldots, m\}](\exists n_i \in \mathbb{N})[\forall s \in \Psi(r_i)]\left(\forall t \in \frac{L}{s}\right)[\| t \| \geq n_i \Rightarrow D_I]$$

where the diagnosability condition D_I is

$$w \in [P_L^{-1} P(st)] \Rightarrow [F_i^{IR} \in \ell^R(w)] \vee [F_i^{IP} \in \ell^R(w)]$$

It is possible to give an interpretation of Type-I diagnosability similar to that of Type-O diagnosability. For the sake of convenience, the label F_i^I shall be used to denote either F_i^{IR}

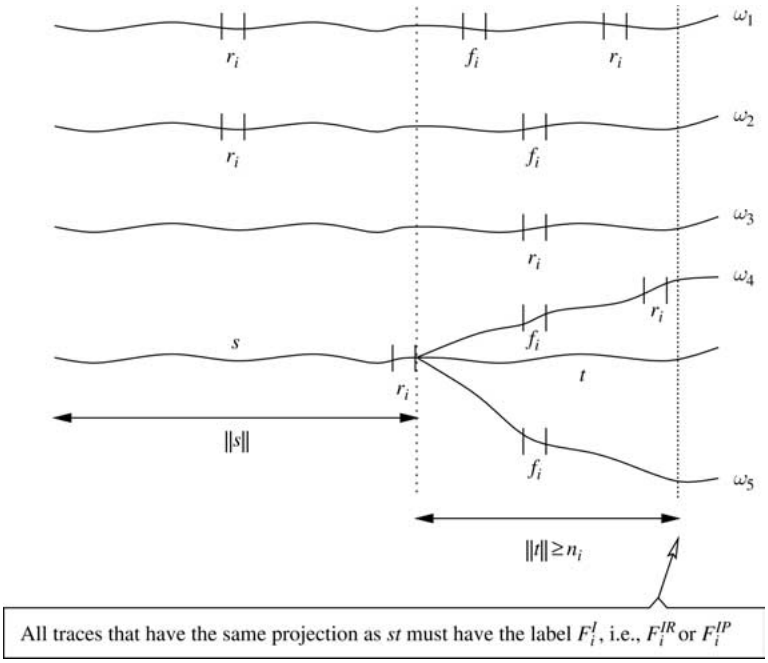


Figure 7. Type-I diagnosability.

or F_i^{IP}. The key difference between the two notions is the following. In Type-O diagnosability, one can assert the occurrence of a fault event f_i, but one cannot be certain as to whether or not f_i has reset, or whether f_i has reset and reoccurred once or many times. In Type-I diagnosability, one can assert the occurrence and reset of a fault event f_i, but one cannot be certain as to whether or not f_i has reoccurred once or many times. This fact is depicted in Figure 7. The above difference illustrates the duality between these two notions of diagnosability.

As Type-P and Type-R are stronger notions of diagnosability than Type-O and Type-I, respectively, we would expect that Type-P (respectively, Type-R) diagnosability implies Type-O (Type-I) diagnosability. This is indeed true, and it is a direct consequence of the above definitions.

PROPOSITION 1

i. *Type-P diagnosability* $\Rightarrow$ *Type-O diagnosability.*

ii. *Type-R diagnosability* $\Rightarrow$ *Type-I diagnosability.*

Figure 8 summarizes the results of Proposition 1 and the implications of the various notions of diagnosability. Table 1 summarizes the labels introduced in this section and in Section 2.

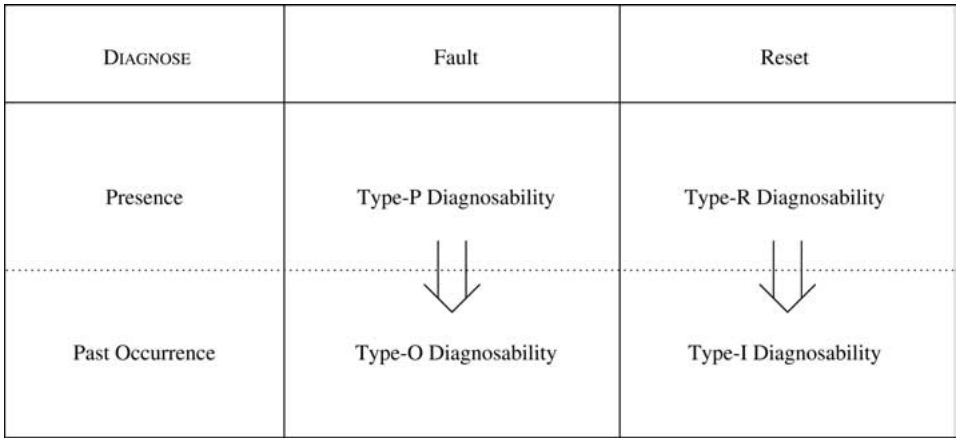


Figure 8. Different notions of diagnosability.

Table 1. Table of labels.

Label	Meaning
F_i	Fault presence without reset occurrence
F_i^{IP}	Fault presence after reset occurrence
F_i^{IR}	Fault reset
F_i^{O}	Fault occurrence (F_i or F_i^{IR} or F_i^{IP})
F_i^{I}	Reset occurrence (F_i^{IR} or F_i^{IP})
F_i^{P}	Fault presence (F_i or F_i^{IP})

Remark 2: The various notions of diagnosability introduced in this section are natural extensions of the notion of diagnosability for permanent faults introduced in Sampath et al. (1995). Intermittent faults evolve dynamically along with the system's evolution. The diagnosability notions introduced here are suitable for dealing with the diagnosis of faults that evolve dynamically. The primary objective is to determine conditions necessary and sufficient to ensure Type-P and Type-R diagnosability. We also wish to determine conditions necessary and sufficient to guarantee the weaker notions of Type-O and Type-I diagnosability.

4. The Diagnoser

The notion of a diagnoser automaton was originally introduced in Sampath et al. (1995). A diagnoser automaton, or simply diagnoser, serves two purposes: (i) on-line detection

and isolation of permanent faults by observing the system behavior; and (ii) off-line analysis of the diagnosability properties of the system regarding permanent faults. The latter is based on an examination of the structure of the diagnoser in order to determine the presence or absence of certain types of cycles termed indeterminate cycles. It turns out that diagnosers are still at the core of the methodology presented in this paper for detecting and isolating intermittent faults, albeit their structure needs to be modified to account for the dynamics of intermittent faults captured by the labeling rules presented in Section 2.2. We denote the modified diagnosers by G_d^R, but continue to refer to them as diagnosers for ease of reading.

The diagnoser G_d^R for G is an automaton

$$G_d^R = (Q_d, \Sigma_o, \delta_d, q_0) \tag{14}$$

where Q_d, Σ_o, δ_d, and q_0 have the usual interpretation. Its initial state q_0 is defined to be $\{(x_0, \{N\})\}$. The transition function δ_d is defined in the same way as in Sampath et al. (1995), but under the new label propagation function LP^R defined below. The state space Q_d is composed of the states of the diagnoser that are reachable from q_0 under δ_d. Therefore, a state q_d of G_d^R is of the form

$$q_d = \{(x_1, \ell_1), \ldots, (x_n, \ell_n)\}$$

where $x_i \in X_o$ and $\ell_i \in \Delta$. Consequently, any label ℓ_i is of the form $\ell_i = \{N\}$ or $\ell_i \subseteq \Delta \backslash \{N\}$, where ℓ_i satisfies the following conditions:

i. if $(F_i \in \ell_i)$ then $(F_i^{IR} \notin \ell_i)$ and $(F_i^{IP} \notin \ell_i)$

ii. if $(F_i^{IR} \in \ell_i)$ then $(F_i \notin \ell_i)$ and $(F_i^{IP} \notin \ell_i)$

iii. if $(F_i^{IP} \in \ell_i)$ then $(F_i \notin \ell_i)$ and $(F_i^{IR} \notin \ell_i)$

Since faults evolve dynamically the rules governing label propagation are different from those of Sampath et al. (1995). These rules are specified by the label propagation function LP^R, which is defined as follows.

DEFINITION 7 *The label propagation function is denoted by LP^R. $LP^R: X_o \times \Delta \times \Sigma_o^* \to \Delta$. Given $x \in X_o$, $\ell \in \Delta$, and $s \in L_o(G, x)$, LP^R propagates the label ℓ over s starting from x and following the dynamics of G and the rules of the label function ℓ^R. It is defined as follows:*

i. $LP^R(x, \ell, s) = \{N\}$ *if* $\forall i, (\ell = \{N\}) \wedge (f_i \notin s) \wedge (r_i \notin s)$;

ii. $\forall i \in \{1, \ldots, m\}$,

$$
\begin{array}{rlll}
F_i \in LP^R(x,\ell,s) & \textit{if} & 1. & (F_i \in \ell) \wedge (r_i \notin s), \text{ or} \\
 & & 2. & (F_i^{IR} \notin \ell) \wedge (F_i^{IP} \notin \ell) \wedge (f_i \in s) \wedge (r_i \notin s), \\
F_i^{IR} \in LP^R(x,\ell,s) & \textit{if} & 1. & (F_i^{IR} \in \ell) \wedge [\ell^R(s) = \{N\} \vee \ell^R(s) = \{F_i^{IR}\}], \\
 & & 2. & [(F_i \in \ell) \vee (F_i^{IP} \in \ell)] \wedge \ell^R(s) = \{F_i^{IR}\}, \textit{ or} \\
 & & 3. & \ell = \{N\} \wedge \ell^R(s) = \{F_i^{IR}\}, \\
F_i^{IP} \in LP^R(x,\ell,s) & \textit{if} & 1. & (F_i^{IP} \in \ell) \wedge [\ell^R(s) = \{N\} \vee \ell^R(s) = \{F_i\} \vee \\
 & & & \qquad \ell^R(s) = \{F_i^{IP}\}], \\
 & & 2. & (F_i^{IR} \in \ell) \wedge [\ell^R(s) = \{F_i\} \vee \ell^R(s) = \{F_i^{IP}\}], \textit{ or} \\
 & & 3. & [\ell = \{N\} \ \vee \ \ell = \{F_i\}] \wedge \ell^R(s) = \{F_i^{IP}\}
\end{array}
$$

Note that Definition 7 is consistent with Definition 1. The behavior of LP^R is illustrated below by four different cases (not exhaustive). Let $x' \in S(x,\sigma)$ with $\delta(x,s\sigma) = x'$ and let ℓ' be the label associated with x' obtained by propagating ℓ associated with x:

a. if $\ell = \{N\}$ and s contains no fault events, then the label ℓ' is also $\{N\}$.

b. if $\ell = \{F_i, F_j, F_k^{IP}\}$ and s contains no fault events, then the label ℓ' is also $\{F_i, F_j, F_k^{IP}\}$.

c. if $\ell = \{N\}$ and s contains fault events f_i, f_j, f_k, and reset events r_j, r_k with the reset event r_j and the fault event f_k being the last ones to occur among $\{f_j, r_j\}$ and $\{f_k, r_k\}$, respectively, then $\ell' = \{F_i, F_j^{IR}, F_k^{IP}\}$.

d. if $\ell = \{F_i, F_j^{IR}, F_k^{IP}\}$ and s contains the two fault events f_j and f_ℓ, and the two reset events r_i and r_k, then $\ell' = \{F_i^{IR}, F_j^{IP}, F_k^{IR}, F_\ell\}$.

We state a few properties of the diagnoser that follow directly from its construction.

Property 1: Let $q \in Q_d$. Then

$$(x_1,\ell_1),(x_2,\ell_2) \in q \Leftrightarrow \exists\, s_1, s_2 \in L$$

such that

$$s_{1f}, s_{2f} \in \Sigma_o, \delta(x_0,s_1) = x_1, \delta(x_0,s_2) = x_2$$

and

$$P(s_1) = P(s_2)$$

Property 2: Let $q_1, q_2 \in Q_d$ and $s \in \Sigma^*$ such that $(x_1,\ell_1) \in q_1, (x_2,\ell_2) \in q_2, \delta_d[q_1, P(s)] = q_2$. Then

$$(F_i^O \notin \ell_2) \Rightarrow (F_i^O \notin \ell_1)$$

Property 3: Let $q_1, q_2 \in Q_d$ and $s \in \Sigma^*$ such that $(x_1, \ell_1) \in q_1$, $(x_2, \ell_2) \in q_2$, $\delta_d[q_1, P(s)] = q_2$. Then

$$(F_i^I \notin \ell_2) \Rightarrow (F_i^I \notin \ell_1)$$

In summary, the diagnoser G_d^R is constructed as follows. Let the current state of the diagnoser be q_1, and let the next observed event be σ. The new state of the diagnoser q_2 is computed via the following three-step process:

1. For every state estimate x in q_1, compute the reach due to σ, given by $S(x, \sigma) = \{\delta(x, s\sigma) : s \in \Sigma_{uo}^*$ and $\delta(x, s\sigma)$ is defined in $G\}$.

2. Let $x' \in S(x, \sigma)$ with $\delta(x, s\sigma) = x'$. Propagate the label ℓ associated with x according to Definition 7 to obtain the label ℓ' associated with x'.

3. Let q_2 be the set of all (x', ℓ') pairs computed following the previous steps, for each (x, ℓ) in q_1.

Figure 9 presents an example of the construction of a diagnoser. The set of observable events in $\Sigma_o = \{\alpha, \beta, \lambda\}$. They are two faults types, F_1 and F_2, with corresponding fault and reset events $\{f_1, r_1\}$ and $\{f_2, r_2\}$, respectively. The initial state of G_d^R is $\{(1, \{N\})\}$, denoted by $1N$ in the figure for the sake of simplicity. Similar compact notation is used in all the figures in this paper. The effect of the new label propagation function LP^R, as compared to the function LP in Sampath et al. (1995), manifests itself when state 8 is first reached after observed trace $\alpha\beta$, yielding the label F_1^{IR} due to the r_1 transition between

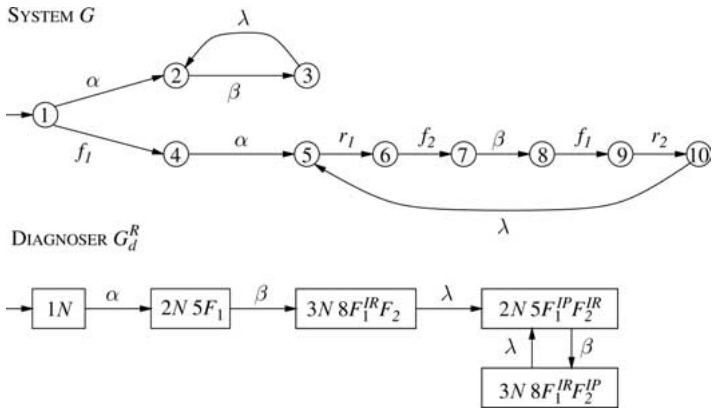


Figure 9. Diagnoser construction.

states 5 and 6. As the system settles in the cycle 5–6–7–8–9–10–5, the diagnoser will alternate between the states $\{(2,\{N\}),(5,\{F_1^{IP},F_2^{IR}\})\}$ and $\{(3,\{N\}),(8,\{F_1^{IR},F_2^{IP}\})\}$.

5. Necessary and Sufficient Conditions for Type-P and Type-R Diagnosability

We present necessary and sufficient conditions for a language L to be Type-P diagnosable or Type-R diagnosable.

5.1. Preliminaries

We define the notions of $F_i^P(F_i^{IR})$-certain states, $F_i^P(F_i^{IR})$-uncertain states, and $F_i^P(F_i^{IR})$-indeterminate cycles in a way similar to Sampath et al. (1995).

DEFINITION 8

1. *A state $q\in Q_d$ is said to be F_i^P-certain if $\forall(x,\ell)\in q, F_i^P\in\ell$.*

2. *A state $q\in Q_d$ is said to be F_i^{IR}-certain if $\forall(x,\ell)\in q, F_i^{IR}\in\ell$.*

3. *A state $q\in Q_d$ is said to be F_i^P-uncertain if $\exists(x,\ell),(y,\ell')\in q$, such that $F_i^P\in\ell$ and $F_i^P\notin\ell'$.*

4. *A state $q\in Q_d$ is said to be F_i^{IR}-uncertain if $\exists(x,\ell),(y,\ell')\in q$, such that $F_i^{IR}\in\ell$ and $F_i^{IR}\notin\ell'$.*

We will say that a state is non-F_i^x-certain if it is not F_i^x-certain; note that non-F_i^x-certain does not imply F_i^x-uncertain.

DEFINITION 9 *A set of states $x_1,x_2,\ldots,x_n\in X$ is said to form a cycle in G if $\exists s\in L(G,x_1)$ such that $s=\sigma_1\sigma_2\ldots\sigma_n$ and $\delta(x_\ell,\sigma_\ell)=x_{(\ell+1)\bmod n}, \ell=1,2,\ldots,n$.*

DEFINITION 10 *F_i^P-indeterminate cycle*
A set of non-F_i^P-certain states $q_1,q_2,\ldots,q_n\in Q_d$, with at least one F_i^P-uncertain state, is said to form an F_i^P-indeterminate cycle if the following condition (C1) is satisfied.

C1. States $q_1,q_2,\ldots,q_n\in Q_d$ form a cycle in G_d^R with $\delta_d(q_u,\sigma_u)=q_{u+1}$, $u=1,\ldots,n-1, \delta_d(q_n,\sigma_n)=q_1$ where $\sigma_u\in\Sigma_o, u=1,\ldots,n$.

Considering the states $q_1,q_2,\ldots,q_n\in Q_d, \exists\,(x_u^k,\ell_u^k)\in q_u, u=1,\ldots,n$, and $k=1,\ldots,m$ such that:

$(F_i^P\in\ell_u^k)$ for some u and k

and the sequence of states $\{x_u^k\}, u = 1, \ldots, n, k = 1, \ldots, m$, forms a cycle in G' with

$$(x_u^k, \sigma_u, x_{(u+1)}^k) \in \delta_{G'}, \quad u = 1, \ldots, n-1, k = 1, \ldots, m$$
$$(x_n^k, \sigma_n, x_1^{k+1}) \in \delta_{G'}, \quad k = 1, \ldots, m-1$$

and

$$(x_n^m, \sigma_n, x_1^1) \in \delta_{G'},$$

DEFINITION 11 *F_i^{IR}-indeterminate cycle*
A set of non-F_i^{IR}-certain states $q_1, q_2, \ldots, q_n \in Q_d$, with at least one F_i^{IR}-uncertain state, is said to form an F_i^{IR}-indeterminate cycle if the following condition (C2) is satisfied.

C2. States $q_1, q_2, \ldots, q_n \in Q_d$ form a cycle in G_d^R with $\delta_d(q_u, \sigma_u) = q_{u+1}, u = 1, \ldots, n-1, \delta_d(q_n, \sigma_n) = q_1$ where $\sigma_u \in \Sigma_o$, $u = 1, \ldots, n$.

Considering the states $q_1, q_2, \ldots, q_n \in Q_d, \exists\, (x_u^k, \ell_u^k) \in q_u, u = 1, \ldots, n$, and $k = 1, \ldots, m$ such that:

$$(F_i^{IR} \in \ell_u^k) \quad \text{for some } u \text{ and } k$$

and the sequence of states $\{x_u^k\}, u = 1, \ldots, n, k = 1, \ldots, m$, forms a cycle in G' with

$$(x_u^k, \sigma_u, x_{(u+1)}^k) \in \delta_{G'}, \quad u = 1, \ldots, n-1, k = 1, \ldots, m$$
$$(x_n^k, \sigma_n, x_1^{k+1}) \in \delta_{G'}, \quad k = 1, \ldots, m-1$$

and

$$(x_n^m, \sigma_n, x_1^1) \in \delta_{G'}$$

The proof of the following result is straight forward and therefore omitted.

PROPOSITION 2 *If a language L is Σ_f-recurrent and Σ_r-recurrent, then every cycle in any system model G, such that $\mathscr{L}(G) = L$, has the following properties:*

1. *The cycle contains the fault event f_i iff it contains the reset event r_i.*

2. *Consider any trace $\omega = s_1 s_2 s_3$ in L where $s_1 \in \Sigma^*$, $s_3 \in \Sigma^*$, and $\exists x_1, x_2, \ldots, x_n \in X, s_2 \in L(G, x_1)$, $s_2 = (\sigma_1 \sigma_2 \cdots \sigma_n)^m \sigma_1 \sigma_2 \cdots \sigma_p$ and $\delta(x_\ell, \sigma_\ell) = x_{(\ell+1) \bmod n}$, $\ell = 1, 2, \ldots, n$, $p \leq n$, and $m, n, p \in \mathbb{N}$ (i.e., s_2 directly enters a cycle in the state transition diagram G and completes at least one loop of this cycle). If $f_i \in s_1$ then $f_i \in s_2$ and $r_i \in s_2$ (i.e., any cycle that occurs after the occurrence of a fault necessarily contains the fault and the reset).*

5.2. Main Results

The following assumptions (A1) and (A2), together with Σ_f- and Σ_r-recurrence (equations (12) and (13)), are critical for the development of necessary and sufficient conditions to ensure Type-P and Type-R diagnosability.

A1. $\forall \omega = s_1 y^n s_2 \in L,\ s_1,\ s_2 \in \Sigma^*,\ n \in \mathbb{N}$, s.t. $f_i, r_i \in y, y$ satisfies the following: $y = y_1 f_i y_2 \sigma_o y_3 r_i y_4$ where $\sigma_o \in \Sigma_o,\ y_1,\ y_4 \in \Sigma^*,\ y_2,\ y_3 \in (\Sigma_o \cup \Sigma_{uo}) \setminus \{f_i, r_i\}$.

A2. $\forall \omega = s_1 y^n s_2 \in L,\ s_1,\ s_2 \in \Sigma^*,\ n \in \mathbb{N}$, s.t. $f_i, r_i \in y, y$ satisfies the following: $y = y_1 r_i y_2 \sigma_o y_3 f_i y_4$ where $\sigma_o \in \Sigma_o,\ y_1,\ y_4 \in \Sigma^*,\ y_2,\ y_3 \in (\Sigma_o \cup \Sigma_{uo}) \setminus \{f_i, r_i\}$.

Assumption (A1) [resp.(A2)] implies that for any cycle in G, there exists at least one observable event between at least one pair (f_i, r_i)[resp.(r_i, f_i)]. This excludes the possibility of having a cycle with only unobservable events between all pairs $(f_i, r_i)[(r_i, f_i)]$, which would have prevented the label F_i^{IP}[resp.F_i^{IR}] from appearing in the corresponding cycle of the diagnoser G_d^R and therefore prevented the detection of any instance where the fault [resp.reset] is present.

The results that follow provide conditions necessary and sufficient to ensure Type-P and Type-R diagnosability.

THEOREM 1 *Type-P diagnosability*
Consider the language L generated by automaton G. Assume that L is Σ_f- and Σ_r-recurrent and satisfies (A1). *L is Type-P diagnosable iff there are no F_i^P-indeterminate cycles in the diagnoser G_d^R.*

Proof:

- Necessity:
  We must prove that if the language L is Type-P diagnosable then there are no F_i^P-indeterminate cycles. We prove the contrapositive, namely, that if there are F_i^P-indeterminate cycles in the diagnoser G_d^R then the language L is not Type-P diagnosable. Assume there exist states $q_1, q_2, \ldots, q_n \in Q_d$ that form an F_i^P-indeterminate cycle and $\delta_d(q_i, \sigma_i) = q_{(i+1) \bmod n}$. Consider the sequence of state components $(x_u^k, \ell_u^k) \in q_u, u = 1, \ldots, n, k = 1, \ldots, m$ with $m \in \mathbb{N}$, that forms a cycle in G' with $F_i^P \in \ell_{u''}^{k''}$, for some $u'' \in \{1, \ldots, n\}$ and some $k'' \in \{1, \ldots, m\}$. By Definition 10 a sequence of states with the above feature exists. Also Definition 10 directly implies that $q_1, q_2, \ldots, q_n$ are non-F_i^P-certain states with at least one F_i^P-uncertain state. In other words, there exists at least one state $q_{u'}, u' \in \{1, \ldots, n\}$, such that $(x_{u'}^{k'}, \ell_{u'}^{k'}) \in q_{u'}$ and $F_i^P \in \ell_{u'}^{k'}, k' \in \{1, \ldots, m\}$. Then we have

$$\delta(x_u^k, s_u^k \sigma_u) = x_{(u+1)}^k, \quad u = 1, \ldots, n-1, k = 1, \ldots, m$$
$$\delta(x_n^k, s_n^k \sigma_n) = x_1^{k+1}, \quad k = 1, \ldots, m-1$$

and

$$\delta(x_n^m, s_n^m \sigma_n) = x_1^1$$

where

$$s_u^k \in L(G, x_u^k) \text{ and } s_u^k \in \Sigma_{uo}^*$$

By Proposition 2, since $F_i^P \in \ell_{u'}^{k'}, f_i \in s_u^k$ for some $u \in \{1, \ldots, n\}$ and $k \in \{1, \ldots, m\}$. Let $s \in \Psi(f_i)$ such that

$$\delta(x_0, s) = x_{u'}^{k'}$$

Let $t \in L/s$ be arbitrarily long such that $\delta_d(q_0, st) = q_u, u \in \{1, \ldots, n\}$, and let v be a subtrace of st such that $\delta_d(q_{u''}, v) = q_{u''}, u'' \in \{1, \ldots, n\}$. Note that the presence of the subtrace v implies that the events contained in v, and therefore in st, cycle at least once between states q_1 to q_n of the diagnoser. Pick any $t' \leq t$. Then $\delta_d(q_0, st') = q_u$ for some $u \in \{1, \ldots, n\}$. Since $q_1, q_2, \ldots, q_n$ are non-F_i^P-certain states, $\forall t', \exists \omega \in P_L^{-1}[P(st')]$ such that $F_i^P \notin \ell^R(\omega)$. Therefore, the chosen s violates the definition of Type-P diagnosability. Hence L is not Type-P diagnosable.

- Sufficiency:
  We must prove that if there are no F_i^P-indeterminate cycles then the language L is Type-P diagnosable. We prove the contrapositive, namely, that if the language L is not Type-P diagnosable then there exists at least one F_i^P-indeterminate cycle. Assume that the language L is not Type-P diagnosable. Then, from Definition 3, this implies that

$$[\exists i \in \{0, \ldots, m'\}](\forall n_i \in \mathbb{N})[\exists s \in \Psi(f_i)]\left(\exists t \in \frac{L}{s}\right)[\| t \| \geq n_i \wedge \neg D] \tag{15}$$

  where the ''non-diagnosability'' condition $\neg D$ is

$$\forall t' \leq t\colon \exists w \in [P_L^{-1} P(st')](F_i^P \notin \ell^R(w)) \tag{16}$$

  Consider the above $s \in \Psi(f_i)$. Since G and G_d^R are finite-state automata and L is Σ_f-recurrent and Σ_r-recurrent, we can find n_i in equations (12) and (13) such that if $\| t \| > n_i$ then the following five properties hold (see Figure 10).

  i. $f_i \in t$

  ii. st leads to a cycle in G' through state components x_u^k, where $(x_u^k, \ell_u^k) \in q_u, u = 1, \ldots, n$, and $k = 1, \ldots, m$, form a cycle in G' with

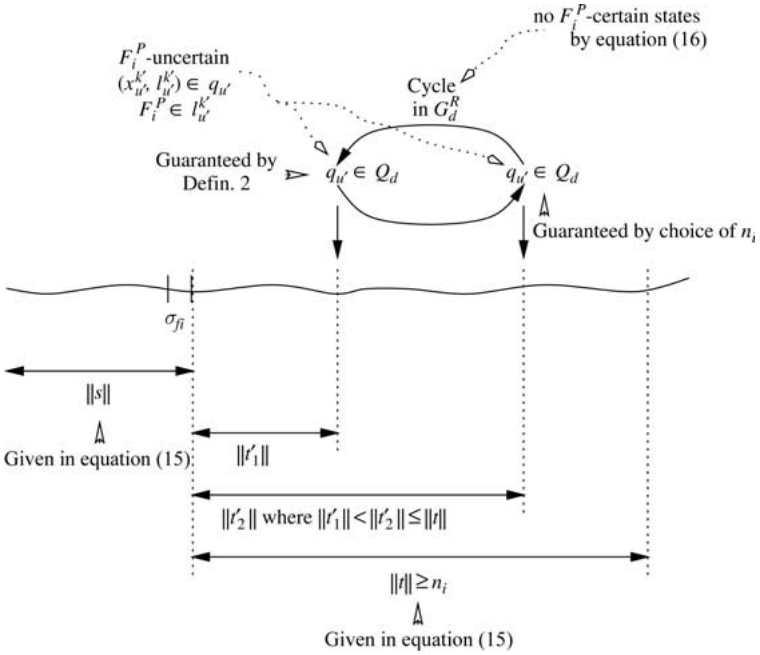


Figure 10. The sufficiency part of the proof of Theorem 1.

$$(x_u^k, \sigma_u, x_{(u+1)}^k) \in \delta_{G'}, \qquad u = 1, \ldots, n-1, k = 1, \ldots, m$$
$$(x_n^k, \sigma_n, x_1^{k+1}) \in \delta_{G'}, \qquad k = 1, \ldots, m-1$$

and

$$(x_n^m, \sigma_n, x_1^1) \in \delta_{G'}$$

iii. *st* leads to a corresponding cycle in G_d^R, that is, there exist states $q_1, q_2, \ldots, q_n \in Q_d$ that form a cycle in G_d^R with

$$\delta_d(q_u, \sigma_u) = q_{u+1}, u = 1, \ldots, n-1, \text{ and}$$
$$\delta_d(q_n, \sigma_n) = q_1 \text{ where } \sigma_u \in \Sigma_o, u = 1, \ldots, n$$

iv. Because of (A1), there exists a prefix st'_1 of *st* such that:
- $\delta(x_0, st'_1) = x_{u'}^{k'}, u' \in \{1, \ldots, n\}, k' \in \{1, \ldots, m\}$, where $(x_{u'}^{k'}, \ell_{u'}^{k'}) \in q_{u'}$ and
- $F_i^P \in \ell_{u'}^{k'}$

v. As a result of (A1), there exists a prefix st'_2 of *st* such that:
- st'_1 is a prefix of st'_2, i.e., $\| t'_1 \| < \| t'_2 \| \leq \| t \|$,
- $\delta(x_0, st'_2) = \delta(x_0, st'_1) = x_{u'}^{k'}$, where $(x_{u'}^{k'}, \ell_{u'}^{k'}) \in q_{u'}$ and
- $F_i^P \in \ell_{u'}^{k'}$.

Thus state $q_{u'}$ is F_i^P-uncertain. Furthermore, equation (16) directly implies that none of the states $q_1, q_2, \ldots, q_n$ are F_i^P-certain. Therefore the cycle $q_1, q_2, \ldots, q_n \in Q_d$ is an F_i^P-indeterminate cycle. ■

THEOREM 2 *Type-R diagnosability*
Consider the language L generated by automaton G. Assume that L is Σ_f- *and* Σ_r-*recurrent and satisfies* (A2). *L is Type-R diagnosable iff there are no* F_i^{IR}-*indeterminate cycles in the diagnoser* G_d^R.

The proof of Theorem 2 is omitted as it is similar to that of Theorem 1.

COROLLARY 1 *Suppose that L is* Σ_f- *and* Σ_r-*recurrent and satisfies* (A1), (A2), *and equations* (12) *and* (13). *Then L is Type-P diagnosable iff it is Type-R diagnosable*.

Proof: Assume that L is not Type-P diagnosable. Then there exists a set of states $q_1, q_2, \ldots, q_n \in Q_d$ that form an F_i^P-indeterminate cycle. By Proposition 2, (A1) and (A2), $q_1, q_2, \ldots, q_n$ is also a set of non-F_i^{IR}-certain states with at least one F_i^{IR}-uncertain state. Furthermore since $F_i^P \in \ell_u^k$ for some u and k, $u = 1, \ldots, n$, and $k = 1, \ldots, m$, then $F_i^{IR} \in \ell_{u'}^{k'}$ for some $u', k', u' = 1, \ldots, n, k' = 1, \ldots, m$. Therefore, the states $q_1, q_2, \ldots, q_n \in Q_d$ form an F_i^{IR}-indeterminate cycle. Hence L is not Type-R diagnosable.

By arguments similar to the above we can show that if the system is not Type-R diagnosable then it is not Type-P diagnosable. ■

Remark 3: Corollary 1 is not true if Assumption (A1) or (A2) is relaxed.

We conclude this section by observing that Assumptions (A1) and (A2), along with the Σ_f- and Σ_r-recurrent assumptions, are only sufficient but not necessary in the proofs of Theorems 1 and 2, respectively. We record this observation in the following corollary.

COROLLARY 2 *Consider the language L generated by automaton G.*

i. There are no F_i^{IP}-*indeterminate cycles in* G_d^R *if L is Type-P diagnosable.*

ii. There are no F_i^{IR}-*indeterminate cycles in* G_d^R *if L is Type-R diagnosable.*

5.3. Examples Illustrating the Results

Figures 11 and 12 present examples of systems with Σ_f- and Σ_r-recurrent languages that satisfy all assumptions, including (A1) and (A2). The system in Figure 11 is neither Type-P diagnosable nor Type-R diagnosable. The set of states 5, 7 and 9, 11 both form cycles in G' (not shown in figure). States q_1 and q_2 in G_d^R, which respectively include 7, 11 and 5, 9 as components, form a cycle in G_d^R.

In the system of Figure 12, states q_1, q_2, and q_3 form a cycle in G_d^R and are non-F_i^P-certain. State q_2 is the only F_i^P-uncertain state of the cycle. However, there are no cycles in G' corresponding to the sequence of state components 10–12–13. Therefore, there are no

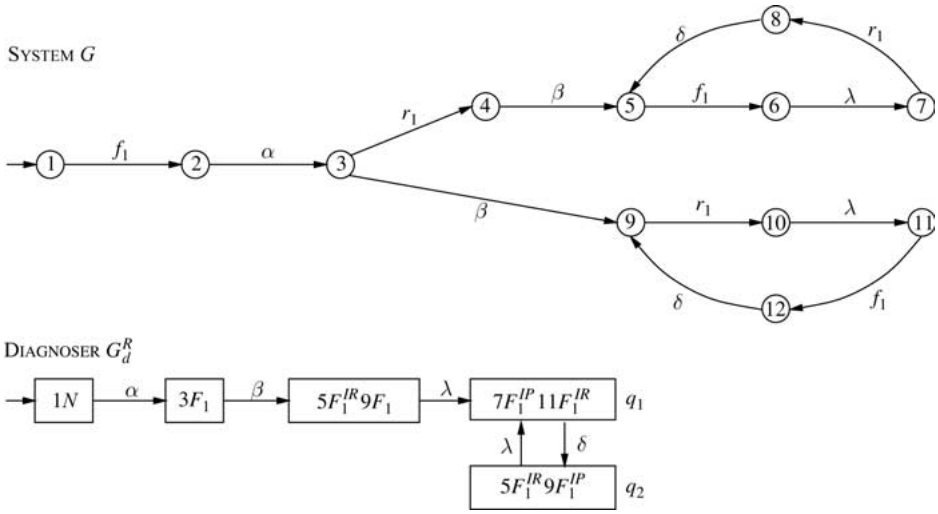


Figure 11. Example of a system that is not Type-P and not Type-R diagnosable.

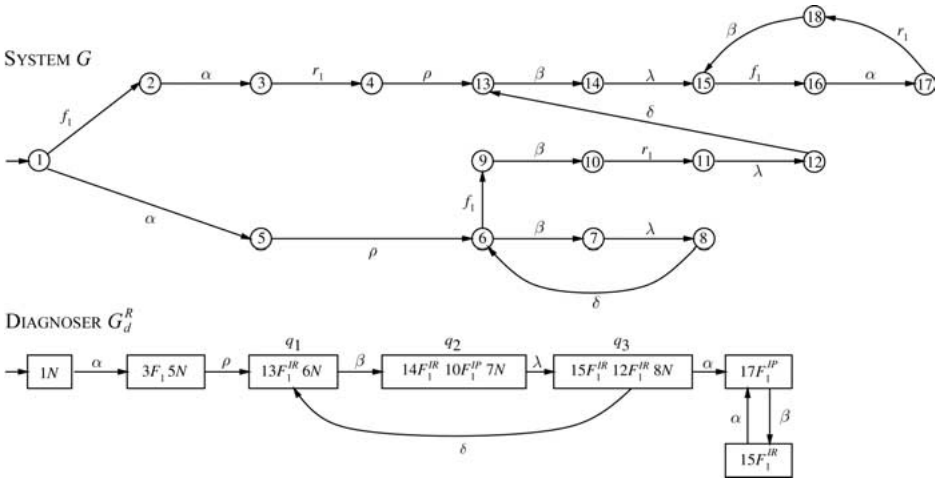


Figure 12. Example of a Type-P and Type-R diagnosable system.

F_i^P-indeterminate cycles and the language L is Type-P diagnosable. By Corollary 1 it is also Type-R diagnosable.

6. Necessary and Sufficient Conditions for Type-O and Type-I Diagnosability

We present necessary and sufficient conditions for a language L to be Type-O diagnosable or Type-I diagnosable.

6.1. Preliminaries

DEFINITION 12

1. *A state $q \in Q_d$ is said to be F_i^O-certain if $\forall (x, \ell) \in q, F_i^O \in \ell$.*

2. *A state $q \in Q_d$ is said to be F_i^I-certain if $\forall (x, \ell) \in q, F_i^I \in \ell$.*

3. *A state $q \in Q_d$ is said to be F_i^O-uncertain if $\exists (x, \ell), (y, \ell') \in q$, such that $F_i^O \in \ell$ and $F_i^O \notin \ell'$.*

4. *A state $q \in Q_d$ is said to be F_i^I-uncertain if $\exists (x, \ell), (y, \ell') \in q$, such that $F_i^I \in \ell$ and $F_i^I \notin \ell'$.*

DEFINITION 13 *F_i^O-indeterminate cycle*
A set of F_i^O-uncertain states $q_1, q_2, \ldots, q_n \in Q_d$ is said to form an F_i^O-indeterminate cycle if the following condition (C3) *is satisfied.*

C3. States $q_1, q_2, \ldots, q_n \in Q_d$ form a cycle in G_d^R with $\delta_d(q_u, \sigma_u) = q_{u+1}, u = 1, \ldots, n-1, \delta_d(q_n, \sigma_n) = q_1$ where $\sigma_u \in \Sigma_o, u = 1, \ldots, n$.

Considering the states $q_1, q_2, \ldots, q_n \in Q_d, \exists (x_u^k, \ell_u^k), (y_u^r, \tilde{\ell}_u^r) \in q_u, u = 1, \ldots, n, k = 1, \ldots, m$, and $r = 1, \ldots, m'$ such that:

$$[(F_i^O \in \ell_u^k) \wedge (F_i^O \notin \tilde{\ell}_u^r)], \text{ for all } u, k, \text{and } r \tag{17}$$

and the sequences of states $\{x_u^k\}, u = 1, \ldots, n, k = 1, \ldots, m$, and $\{y_u^r\}, u = 1, \ldots, n, r = 1, \ldots, m'$, form cycles in G' with

$$(x_u^k, \sigma_u, x_{(u+1)}^k) \in \delta_{G'}, \quad u = 1, \ldots, n-1, k = 1, \ldots, m$$
$$(x_n^k, \sigma_n, x_1^{k+1}) \in \delta_{G'}, \quad k = 1, \ldots, m-1$$

and

$$(x_n^m, \sigma_n, x_1^1) \in \delta_{G'}$$

and

$$(y_u^r, \sigma_u, y_{(u+1)}^r) \in \delta_{G'}, \quad u = 1, \ldots, n-1, r = 1, \ldots, m'$$
$$(y_n^r, \sigma_n, y_1^{r+1}) \in \delta_{G'}, \quad r = 1, \ldots, m'-1$$

and

$$(y_n^{m'}, \sigma_n, y_1^1) \in \delta_{G'}.$$

DEFINITION 14 *F_i^I-indeterminate cycle*
A set of F_i^I-uncertain states $q_1, q_2, \ldots, q_n \in Q_d$ is said to form an F_i^I-indeterminate cycle if the condition (C4) *is satisfied.*

C4. The same as (C3) with the exception that clause (17) is replaced by the following:

$$[(F_i^I \in \ell_u^k) \wedge (F_i^I \notin \tilde{\ell}_u^r)] \text{ for all } u, k, \text{and } r \tag{18}$$

6.2. Main Results

The results that follow provide necessary and sufficient conditions to ensure Type-O and Type-I diagnosability.

THEOREM 3 *Type-O diagnosability*
Consider the language L generated by automaton G. L is Type-O diagnosable iff there are no F_i^O-indeterminate cycles in the diagnoser G_d^R.

Proof:

- Necessity:
  We prove that if a language L is Type-O diagnosable then there are no F_i^O-indeterminate cycles. By contradiction, assume there exist states $q_1, q_2, \ldots, q_n \in Q_d$ such that they form an F_i^O-indeterminate cycle and $\delta_d(q_i, \sigma_i) = q_{(i+1) \bmod n}$. Let $(x_u^k, \ell_u^k), (y_u^r, \tilde{\ell}_u^r) \in q_u$, $u = 1, \ldots, n, k = 1, \ldots, m$, and $r = 1, \ldots, m'$, with $m, m' \in \mathbb{N}$, form corresponding cycles in G' with $F_i^O \in \ell_u^k$ and $F_i^O \notin \tilde{\ell}_u^r$. Then we have

  $$\delta(x_u^k, s_u^k \sigma_u) = x_{(u+1)}^k, \quad u = 1, \ldots, n-1, k = 1, \ldots, m$$
  $$\delta(x_n^k, s_n^k \sigma_n) = x_1^{k+1}, \quad k = 1, \ldots, m-1$$

  and

  $$\delta(x_n^m, s_n^m \sigma_n) = x_1^1$$

  and

  $$\delta(y_u^r, \tilde{s}_u^r \sigma_u) = y_{(u+1)}^r, \quad u = 1, \ldots, n-1, r = 1, \ldots, m'$$
  $$\delta(y_n^r, \tilde{s}_n^r \sigma_n) = y_1^{r+1}, \quad r = 1, \ldots, m'-1$$

  and

  $$\delta(y_n^{m'}, \tilde{s}_n^{m'} \sigma_n) = y_1^1$$

  where

$s_u^k \in L(G, x_u^k), \tilde{s}_u^r \in L(G, y_u^r)$

and

$s_u^k, \tilde{s}_u^r \in \Sigma_{uo}^*$

Since $(x_1^1, \ell_1^1), (y_1^1, \tilde{\ell}_1^1) \in q_1, \exists s_0, \tilde{s}_0 \in L$ such that $\delta(x_0, s_0) = x_1^1$, $\delta(y_0, \tilde{s}_0) = y_1^1$ and $P(s_0) = P(\tilde{s}_0)$ from Property 1. Furthermore, $F_i^O \in \ell_1^1$ implies that $f_i \in s_0$, and $F_i^O \notin \tilde{\ell}_u^r$ means that $f_i \notin \tilde{s}_0$ and $f_i \notin \tilde{s}_u^r$ for all u, r.

Consider the two traces

$$\omega = s_o(s_1^1\sigma_1 s_2^1\sigma_2, \ldots, s_n^1\sigma_n s_1^2\sigma_1 s_2^2\sigma_2, \ldots, s_n^2\sigma_n, \ldots, s_1^m\sigma_1 s_2^m\sigma_2, \ldots, s_n^m\sigma_n)^{km'}$$
$$\tilde{\omega} = \tilde{s}_o(\tilde{s}_1^1\sigma_1 \tilde{s}_2^1\sigma_2, \ldots, \tilde{s}_n^1\sigma_n \tilde{s}_1^2\sigma_1 \tilde{s}_2^2\sigma_2, \ldots, \tilde{s}_n^2\sigma_n, \ldots, \tilde{s}_1^{m'}\sigma_1 \tilde{s}_2^{m'}\sigma_2, \ldots, \tilde{s}_n^{m'}\sigma_n)^{km}$$

for arbitrarily large k. Then $\omega \in L, \tilde{\omega} \in L, P(\omega) = P(\tilde{\omega}) = P(s_0)(\sigma_1\sigma_2 \ldots \sigma_n)^{kmm'}$, and $F_i^O \in \ell^R(\omega)$ since $f_i \in \omega$, while $F_i^O \notin \ell^R(\tilde{\omega})$ since $f_i \notin \tilde{\omega}$. Let $s \in \bar{s_o}$ be such that $s \in \Psi(f_i)$, and let $t \in L/s$ be such that $\omega = st$. By choosing k to be arbitrarily large, we can get $\| t \| > n$ for any given $n \in \mathbb{N}$. Furthermore $\tilde{\omega} \in P_L^{-1}[P(st)]$ and $F_i^O \notin \ell^R(\tilde{\omega})$ since $f_i \notin \tilde{\omega}$. Therefore, the chosen s violates the definition of Type-O diagnosability. Hence L is not Type-O diagnosable.

- Sufficiency:
We prove that if there are no F_i^O-indeterminate cycles then the language L is Type-O diagnosable. Assume that the diagnoser G_d^R does not have any F_i^O-indeterminate cycle, $i = 1, 2, \ldots, m$. For any F_i^O, pick any $s \in L$ such that $s \in \Psi(f_i)$ and let $\delta(x_0, s) = x$. Pick any $t_1 \in L_0(G, x)$. Since we assume there are no cycles of unobservable events in G, there exists $n_0 \in \mathbb{N}$ such that $\| t_1 \| \leq n_0$. Let $\delta(x_0, st_1) = x_1$ and correspondingly in G_d^R, let $\delta_d[q_0, P(st_1)] = q_1$. Then $(x_1, \ell_1) \in q_1$ and $F_i^O \in \ell_1$.

  We distinguish two cases: (I) q_1 is F_i^O-certain and (II) q_1 is F_i^O-uncertain.

  Case I: Suppose q_1 is F_i^O-certain. Then, by Definition 12,

  $(\forall \omega \in P_L^{-1}[P(st_1)])F_i^O \in \ell^R(\omega)$

  Hence L is diagnosable for F_i^O with $n_i = n_0$. Since this is true for any F_i^O, L is diagnosable.

  Case II: Suppose q_1 is F_i^O-uncertain. Consider any $(z, \ell) \in q_1$ such that $F_i^O \in \ell$. We shall then refer to z as an ''x-state'' of q_1. Likewise, if $(z', \ell') \in q_1$ such that $F_i^O \notin \ell'$, then we shall denote z' as a ''y-state'' of q_1. We have assumed that there are no F_i^O-indeterminate cycle in G_d^R. Recalling the definition of an F_i^O-indeterminate cycle, this assumption means that one of the following is true: (i) there are no cycles of F_i^O-uncertain states in G_d^R, or (ii) there exists one or more cycles of F_i^O-uncertain states $q_1, q_2, \ldots, q_n$ in G_d^R but corresponding to any such cycle in G_d^R, there do not exist two sequences $\{x_u^k\}$ *and* $\{y_u^r\}, u = 1, 2, \ldots, n$, and $k, r \in \mathbb{N}$ such that both of them form

cycles in G' ($\{x_u^k\}$ is composed of ''x-states'' of q_u, and $\{y_u^r\}$ is composed of ''y-states'' of $q_u, u = 1, 2, \ldots, n$, cf. Definition 13).

i. Suppose that there are no cycles of F_i^O-uncertain states in G_d^R. Then every F_i^O-uncertain state will lead to an F_i^O-certain state in a bounded number of transitions by Property 2 of label propagation.

ii. Suppose that there exists a cycle C of F_i^O-uncertain states $q_1, q_2, \ldots, q_n$ *in* G_d^R. We show that whenever a fault happens, i.e., when the true state of the system is an ''x-state'', it is not possible to loop for arbitrarily long in this cycle in G_d^R and thereby never detect the fault.

Pick any ''y-state'' $y_u \in q_u$, and let the corresponding label be $\tilde{\ell}_u$. Since $F_i^O \notin \tilde{\ell}_u$, the pair $(y_u, \tilde{\ell}_u) \in q_u$ could only have resulted from a pair $(y_{u-1}, \tilde{\ell}_{u-1}) \in q_{u-1}$ such that $F_i^O \notin \tilde{\ell}_{u-1}$, because of Property 2. That is the ''y-state'' y_u cannot be a successor of any ''x-state'' x_{u-1} along the corresponding trace in G'. Thus, by backward induction, we can always build a cycle of states in G' involving some or all of the ''y-states'' of $q_u, u = 1, 2, \ldots, n$. These ''y-states'' then constitute the sequence y_u^r. Since the cycle of F_i^O-uncertain states $q_1, q_2, \ldots, q_n$ is not F_i^O-indeterminate, there does not exist a cycle in G' involving the ''x-states'' of $q_u, u = 1, 2, \ldots, n$. Hence, if we pick any ''x-state'' x_u in state q_u in the cycle C, then a sufficient long trace $p \in L(G, x_u)$, guaranteed by the liveness assumption, will leave the cycle C of F_i^O-uncertain states, and will lead to an F_i^O-certain state. ■

THEOREM 4 *Type-I diagnosability*
Consider the language L generated by automaton G. L is Type-I diagnosable iff there are no F_i^I-indeterminate cycles in the diagnoser G_d^R.

The proof of Theorem 4 is similar to that of Theorem 3 and is therefore omitted.

7. A Pump-Valve-Controller Example

Consider a small system consisting of mechanical components: a pump, a valve, a controller, and one flow sensor. For the sake of simplicity we assume that the pump and the controller do not fail. The valve has two intermittent fault modes, namely, a stuck- and unstuck-closed fault mode (labelled F_1^x) and a stuck- and unstuck-open fault mode (labelled F_2^x).

In order to construct the complete model G, we first need to form the parallel composition (Pump $\|$ Valve $\|$ Controller) of the individual component models; see Figure 13. The initial states are POFF (Pump OFF), VC (Valve Closed), and C1 (Controller State #1). The meaning of the events and their observable or unobservable status are shown in Table 2. The second step is to combine the sensor map with the synchronized component models (Pump $\|$ Valve $\|$ Controller) and obtain the complete model G. The flow sensor takes either the value F, indicating flow, or the value NF, indicating no flow. The global sensor map is listed in Table 3. This table represents the sensor mapping of the

Table 2. Event list.

C_P: Close_Pump (observable)	O_P: Open_Pump (observable)
NL: No Load (observable)	L: Load (observable)
C_V: Close_Valve (observable) S_C: Stuck_Closed (unobservable) V_C: Valve_Closed (unobservable) US_C: UnStuck_Closed (unobservable)	O_V: Open_Valve (observable) S_O: Stuck_Open (unobservable) V_O: Valve_Open (unobservable) US_O: UnStuck_Open (unobservable)

Table 3. Global sensor map.

$f(PON, VC, \bullet) = NF$	$f(PON, VO, \bullet) = F$
$f(PON, X1, \bullet) = NF$	$f(PON, X2, \bullet) = F$
$f(PON, SC, \bullet) = NF$	$f(PON, SO, \bullet) = F$
$f(PON, X3, \bullet) = NF$	$f(PON, X4, \bullet) = F$
$f(PON, VC_i, \bullet) = NF$	$f(PON, VO_i, \bullet) = F$
$f(PON, X5, \bullet) = NF$	$f(PON, X6, \bullet) = F$
$f(POFF, \bullet, \bullet) = NF$	

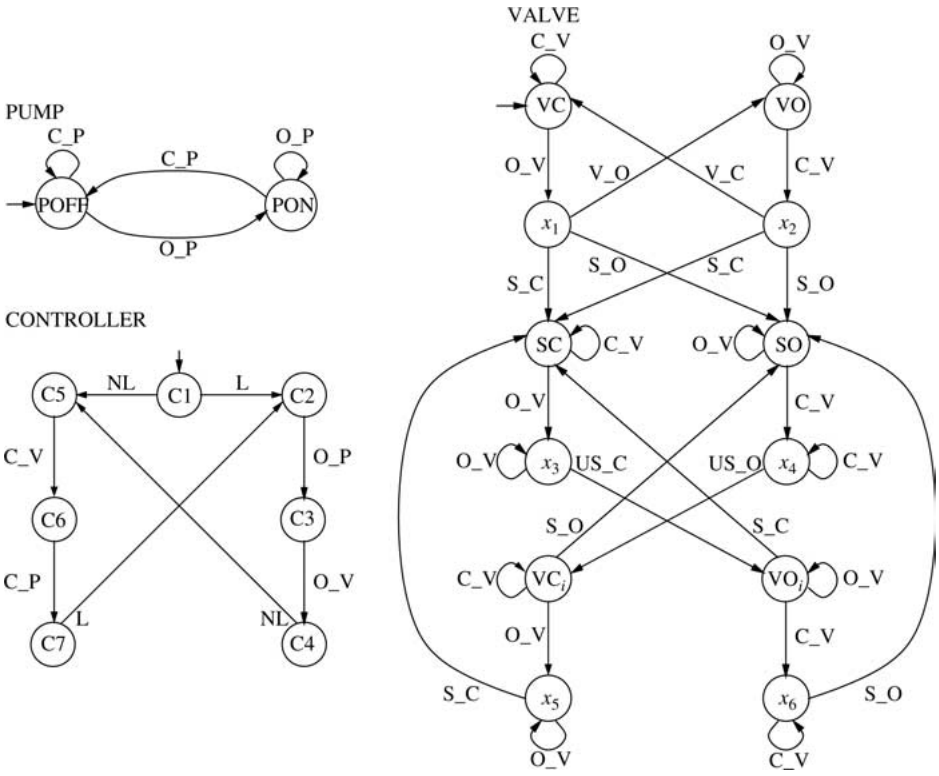


Figure 13. Component models.

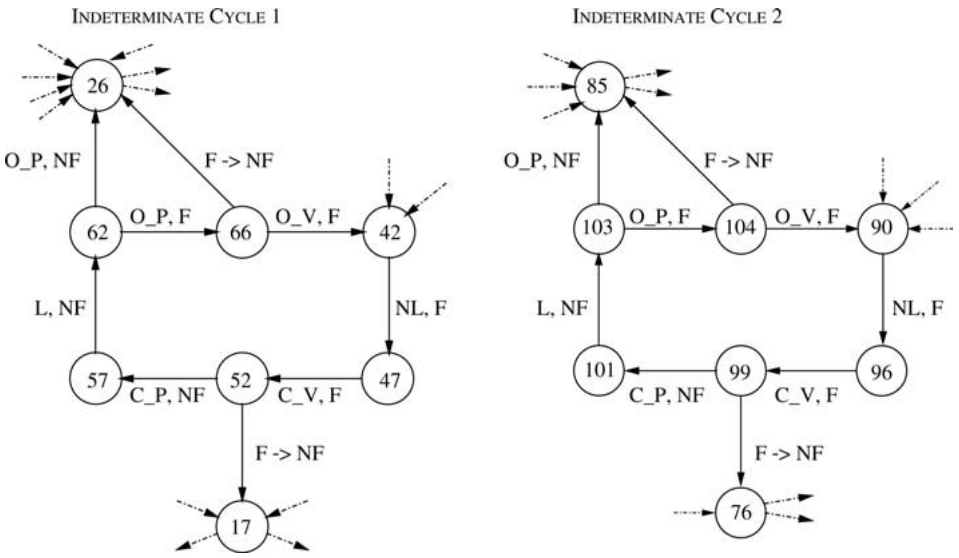


Figure 14. Portions of diagnoser G_d^R showing the indeterminate cycles.

states obtained in the parallel composition (Pump || Valve || Controller). Each state is therefore composed of three ordered state components, for the respective states of the pump, valve, and controller. The symbol • represents any possible state of the given component. The resulting G was obtained using the UMDES-LIB[1] software and has 64 states; it is not depicted here. The next step is to construct the diagnoser G_d^R for G. This is done using UMDES-LIB; G_d^R has 104 states. By inspecting G_d^R, we test the four diagnosability conditions defined in this paper. This analysis shows that there are only two indeterminate cycles in G_d^R and they are both F_2^{IR}-indeterminate cycles. Figure 14 and Tables 4 and 5 describe the two portions of the diagnoser G_d^R that contain the F_2^{IR}-indeterminate cycles. We draw the following conclusions:

- The system is Type-O and Type-I diagnosable since neither F_i^O- nor F_i^I-indeterminate cycles exist.

- No F_i^{IP}-indeterminate cycles exist in $G_d^R, i = 1, 2$. Based on Theorem 1 we need to verify if certain assumptions are satisfied in order to conclude about Type-P diagnosability. The language L is Σ_f- and Σ_r-recurrent and Assumption (A1) is satisfied for both fault types. Therefore the system is Type-P diagnosable.

- Based on Corollary 2 (ii) the fault of type 2 is not Type-R diagnosable since there exist F_2^{IR}-indeterminate cycles in G_d^R. Due to the absence of F_1^{IR}-indeterminate cycles, we need to check for the assumptions mentioned in Theorem 2 in order to conclude about Type-R diagnosability. Assumption (A2) is violated for both fault types and thus despite the absence of F_1^{IR}-indeterminate cycles, no conclusion can be drawn about Type-R diagnosability for faults of type 1.

Table 4. Description of diagnoser states involved in indeterminate cycle 1.

State Number	Concatenation of the Component States
62	$\{(\text{poff,x4,c2})F_2^{IP}, (\text{poff,so,c2})F_2^{IP}, (\text{poff,vci,c2})F_2^{IR}\}$
66	$\{(\text{pon,x4,c3})F_2^{IP}, (\text{pon,so,c3})F_2^{IP}\}$
42	$\{(\text{pon,so,c4})F_2^{IP}\}$
47	$\{(\text{pon,so,c5})F_2^{IP}\}$
52	$\{(\text{pon,x4,c6})F_2^{IP}\}$
57	$\{(\text{poff,x4,c7})F_2^{IP}\}$

Table 5. Description of diagnoser states involved in indeterminate cycle 2.

State Number	Concatenation of the Component States
103	$\{(\text{poff,x4,c2})F_1^{IR}F_2^{IP}, (\text{poff,so,c2})F_1^{IR}F_2^{IP}, (\text{poff,vci,c2})F_1^{IR}F_2^{IR}\}$
104	$\{(\text{pon,x4,c3})F_1^{IR}F_2^{IP}, (\text{pon,so,c3})F_1^{IR}F_2^{IP}\}$
90	$\{(\text{pon,so,c4})F_1^{IR}F_2^{IP}\}$
96	$\{(\text{pon,so,c5})F_1^{IR}F_2^{IP}\}$
99	$\{(\text{pon,x4,c6})F_1^{IR}F_2^{IP}\}$
101	$\{(\text{poff,x4,c7})F_1^{IR}F_2^{IP}\}$

8. Conclusion

Intermittent faults are dynamic in nature, i.e., they can occur, reset, and reoccur repeatedly during a system's operation. They are distinctly different from permanent faults which never reset after they occur. Therefore, system models that capture permanent failures and existing methodologies for diagnosis of permanent failures are not appropriate for modeling and diagnosis of intermittent faults.

We proposed a method for modeling intermittent faults and their resets in the context of discrete event system models. We defined notions of diagnosability that provide information about the status of intermittent faults at different levels of detail. Finally, we developed, via conditions that are necessary and sufficient to ensure different notions of diagnosability, a methodology for diagnosis of intermittent faults.

Our investigation has revealed that: (i) diagnosis of intermittent faults is a problem considerably more intricate than the diagnosis of permanent failures; and (ii) the philosophy and approach to diagnosis of permanent failures developed in Sampath et al. (1995) is powerful and generic, as evidenced by the nature of the results of this paper.

Acknowledgment

This research was supported in part by NSF grant ECS-0080406.

Notes

1. http://www.eecs.umich.edu/umdes/

References

Aghasaryan, A., Fabre, E., Benveniste, A., Boubour, R., and Jard, C. 1998. Fault detection and diagnosis in distributed systems: An approach by partially stochastic Petri nets. *Discrete Event Dynamic Systems: Theory and Applications* 8(2): 203–231.

Benveniste, A., Fabre, E., Haar, S., and Jard, C. 2003. Diagnosis of asynchronous discrete event systems, a net unfolding approach. *IEEE Trans. Automatic Control* 48(5): 714–727.

Bouloutas, A. T. 1990. *Modeling fault management in communication networks*. Ph.D. thesis, Columbia University.

Cassandras, C. G., and Lafortune, S. 1999. *Introduction to Discrete Event Systems*. Kluwer Academic Publishers.

Console, L. 2000. Diagnosis and diagnosability analysis using process algebras. In A. Darwiche and G. Provan (eds.), *Proc. DX'00: Eleventh International Workshop on Principles of Diagnosis*, June, pp. 25–32.

Darwiche, A., and Provan, G. 1996. Exploiting system structure in model-based diagnosis of discrete event systems. In *Pro. Seventh International Workshop on the Principles of Diagnosis, DX-96*, October, Val Morin, Canada.

Dvorak, D., and Kuipers, B. 1992. Model based monitoring of dynamic systems. In W. Hamscher, L. Console, and J. Kleer (eds.), *Readings in Model Based Diagnosis*, Morgan Kaufmann, pp. 249–254.

Debouk, R., Lafortune, S., and Teneketzis, D. 2000. Coordinated decentralized protocols for failure diagnosis of discrete-event systems. *Discrete Event Dynamic Systems: Theory and Applications* 10(1/2): 33–86.

Garcia, E., Morant, F., Blasco-Gimenez, R., Correcher, A., and Quiles, E. 2002. Centralized modular diagnosis and the phenomenon of coupling. In *Proc. IEEE WODES '02 International Workshop on Discrete Event Systems*, October, pp. 161–168.

Hastrudi Zad, S., Kwong, R. H., and Wonham, W. M. 1998. Fault diagnosis in discrete-event systems: Framework and model reduction. In *Proc. 37th IEEE Conf. on Decision and Control*, December, pp. 3769–3774.

Jiang, S., and Kumar, R. 2002. Failure diagnosis of discrete event systems with linear-time temporal logic fault specifications. In *Proc. 2002 American Control Conference*, May, pp. 128–133.

Jiang, S., Kumar, R., and Garcia, H. E. 2002. Diagnosis of repeated failures in discrete event systems. In *Proc. of the 41st IEEE Conference on Decision and Control*, December, pp. 4000–4005.

Lafortune, S., Teneketzis, D., Sampath, M., Sengupta, R., and Sinnamohideen, K. 2001. Failure diagnosis of dynamic systems: An approach based on discrete event systems. In *Proc. 2001 American Control Conference*, June, pp. 2058–2071.

Lamperti, G., and Zanella, M. 1999. Diagnosis of discrete event systems integrating synchronous and asynchronous behavior. In *Proceedings of the Ninth International Workshop on Principles of Diagnosis, DX'99*, pp. 129–139.

Lin, F. 1994. Diagnosability of discrete event systems and its applications. *Discrete Event Dynamic Systems: Theory and Applications* 4(2): 197–212.

Lin, F., Markee, J., and Rado, B. 1993. Design and test of mixed signal circuits: A discrete event approach. In *Proc. 32th IEEE Conf. on Decision and Control*, December, pp. 246–251.

Lunze, J. 2000. Diagnosis of quantised systems. In *IFAC SafeProcess 2000*, June, pp. 28–39.

Pandalai, D. N., and Holloway, L. E. 2000. Template languages for fault monitoring of discrete event processes. *IEEE Transactions on Automatic Control* 45(5): 868–882.

Pencolé, Y. 2000. Decentralized diagnoser approach: Application to telecommunication networks. In A. Darwiche and G. Provan (eds.), *Proc. DX'00: Eleventh International Workshop on Principles of Diagnosis*, June, pp. 185–192.

Pencolé, Y., Cordier, M.-O., and Rozé, L. 2001. A decentralized model-based diagnostic tool for complex systems. In *Proc. of the 13th IEEE International Conf. on Tools with Artificial Intelligence*, November, pp. 95–102.

Provan, G., and Chen, Y.-L. 1998. Diagnosis of timed discrete event systems using temporal causal networks: Modeling and analysis. In *Proc. of the 1998 International Workshop on Discrete Event Systems (WODES'98)*, IEE, August, pp. 152–154.

Provan, G., and Chen, Y.-L. 1999. Model-based diagnosis and control reconfiguration for discrete event systems: An integrated approach. In *Proc. 38th IEEE Conf. on Decision and Control*, December, pp. 1762–1768.

Ramadge, P. J., and Wonham, W. M. 1989. The control of discrete event systems. *Proc. IEEE* 77(1): 81–98.

Sampath, M. 2001. A hybrid approach to failure diagnosis of industrial systems. In *Proc. 2001 American Control Conf.*, June.

Sampath, M., Lafortune, S., and Teneketzis, D. 1998. Active diagnosis of discrete event systems. *IEEE Trans. Automatic Control* 43(7): 908–929.

Sampath, M., Sengupta, R., Sinnamohideen, K., Lafortune, S., and Teneketzis, D. 1995. Diagnosability of discrete event systems. *IEEE Transactions on Automatic Control* 40(9): 1555–1575.

Sampath, M., Sengupta, R., Sinnamohideen, K., Lafortune, S., and Teneketzis, D. 1996. Failure diagnosis using discrete event models. *IEEE Transactions on Control Systems Technology* 4(2): 105–124.

Sengupta, R. 2001. Discrete-event diagnostics of automated vehicles and highways. In *Proc. 2001 American Control Conf.*, June.

Sinnamohideen, K. 2001. Discrete-event diagnostics of heating, ventilation, and air-conditioning systems. In *Proc. 2001 American Control Conf.*, June.

Westerman, G., Kumar, R., Stroud, C., and Heath, J. R. 1998. Discrete event system approach for delay fault analysis in digital circuits. In *Proc. 1998 American Control Conf.*

Williams, B., and Nayak, P. 1996. A model-based approach to reactive self-configuring systems. In *Proceedings of the AAAI*.