Automatica 79 (2017) 315–325

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Diagnosability of intermittent sensor faults in discrete event systems[☆]

Lilian K. Carvalho, Marcos V. Moreira, João Carlos Basilio

Programa de Engenharia Elétrica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, R.J., Brazil

ARTICLE INFO

Article history:
Received 26 April 2015
Received in revised form
16 May 2016
Accepted 10 December 2016

Keywords:
Discrete event system
Automata
Fault diagnosis

ABSTRACT

We address, in this paper, the problem of diagnosing intermittent sensor faults. In order to do so, we employ a model of intermittent loss of observations recently proposed in the literature, and use this model, together with an appropriately modified label automaton, to change the problem of detecting intermittent sensor faults into a problem of diagnosing the language generated by an automaton in the presence of intermittent faults, where the fault event is the unobservable event that models the non-observation of the event whose occurrence is recorded by the sensor subject to intermittent fault. We present necessary and sufficient conditions for diagnosability of intermittent sensor faults and propose two tests to verify intermittent sensor fault diagnosability: the first one based on diagnosers, which can also be used for online diagnosis, and a second one, based on verifiers, which has the advantage of having polynomial time complexity.

1. Introduction

Sensors play a crucial role in the reliability and safety of feedback controlled systems, and their faults have been reported as the cause of several accidents that led to either material or life losses (da Silva, Saxena, Balaban, & Goebel, 2012). It is, therefore, important to find the means to distinguish between sensor malfunction and ordinary (normal) behavior. It is particularly important to check, in practice, if intermittent sensor faults are actually happening with a view to identifying and replacing those sensors that fail frequently (permanently or intermittently) without apparent external causes, and to find out the external causes of the sensor fault (*e.g.*, environmental causes, such as high and low temperatures, pressure, magnetic interference, radiation, *etc.*).

There are basically three main approaches to the problem of detecting incorrect sensor readings (Frank, 1990): (i) simple hardware redundancy with majority voting, (ii) model-based, and (iii) knowledge-based. Hardware redundancy with majority voting is the simplest way to improve sensor reliability; model-based design relies on some model developed for the system under consideration, and the decision regarding the sensor fault occurrence is made based on comparisons between the outputs of the model and of the real system; knowledge-based design employs artificial intelligence techniques such as neural networks and fuzzy logic to develop expert systems. Among the model-based approach, the most relevant works reported in the literature are the incipient work by Clark (1978), who proposed the so-called dedicated observer scheme (DOS), the paper by Frank (1990), which besides presenting a literature survey, also improved the scheme developed by Clark (1978), leading to the so-called generalized observer scheme (GOS), Lunze and Schröder (2004), who proposed a method for the detection and identification of sensor and actuator faults, using discrete event theory, by modeling the plant of the system under consideration as a stochastic automaton, and Ding, Fennel, and Ding (2004), who presented a model-based sensor monitoring scheme for the electronic stability program (ESP) system consisting of an anti-lock break system, a traction control and a yaw torque control. Expert systems were proposed by Athanasopoulou and Chatziathanasiou (2009), who developed an intelligent system for identification and replacement of faulty sensor measurements in thermal power plants, and da Silva et al. (2012), who presented a system for sensor fault diagnosis using neural network approach.

We propose, in this paper, a discrete event approach to the problem of diagnosing intermittent sensor faults by modeling the dynamic system as a deterministic automaton. We assume that the sensor fault diagnosis system is built separately from both the ordinary failure diagnosis and the supervisory control systems, as shown in Fig. 1, and that both the supervisory control and the diagnosis systems can cope with intermittent sensor faults

[☆] The material in this paper was partially presented at the 2013 American Control Conference, June 17–19, 2013, Washington, DC, USA. This paper was recommended for publication in revised form by Associate Editor Jan Komenda under the direction of Editor Christos G. Cassandras.

E-mail addresses: lilian@dee.ufrj.br (L.K. Carvalho), moreira@dee.ufrj.br (M.V. Moreira), basilio@dee.ufrj.br (J.C. Basilio).

http://dx.doi.org/10.1016/j.automatica.2017.01.017
0005-1098/© 2017 Elsevier Ltd. All rights reserved.

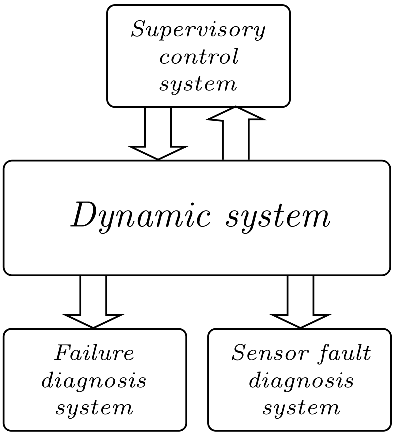

Fig. 1. Schematic diagram showing the supervisory control, the failure diagnosis system, and the fault diagnosis systems.

(Alves, Basilio, da Cunha, Carvalho, & Moreira, 2014; Carvalho, Basilio, & Moreira, 2012). In the proposed structure, the supervisory control and diagnosis systems, being tolerant to intermittent sensor faults, allow the system to continue working properly when such faults occur, whereas the sensor fault diagnosis system detects the occurrence of sensor faults. We employ the model for intermittent loss of observations recently proposed by Carvalho et al. (2012), and convert the problem of detecting intermittent sensor faults into a problem of diagnosing intermittent failure. In this regard, we present necessary and sufficient conditions for intermittent sensor fault diagnosability and propose two tests to verify intermittent sensor fault diagnosability: the first one is based on diagnosers, which can also be used for online diagnosis, and the second one which is based on verifiers has the advantage of having polynomial time complexity. It is worth remarking that, for the sensor fault diagnosis system, any failure event that may appear in the model will be treated as an ordinary unobservable event.

The problem considered in this paper has several differences from that solved by Contant, Lafortune, and Teneketzis (2004), which addressed the problem of diagnosing intermittent failure, namely that: (i) there is no reset event here; (ii) cyclic paths with unobservable events are allowed here, as opposed to Contant et al. (2004), which prevent the existence of cyclic paths. Our approach is also different from that by Thorsley, Yoo, and Garcia (2008), who addressed the problem of stochastic discrete event systems under unreliable observation, and also from that by Ushio and Takai (2009) in the context of supervisory control, which modeled the unreliable observations using masks.

Sensor faults have also been addressed in the context of supervisory control (Alves et al., 2014; Rohloff, 2005; Sanchez & Montoya, 2006; Ushio & Takai, 2009; Xu & Kumar, 2009), and as part of the design requirements of fault diagnosis systems (Carvalho et al., 2012; Carvalho, Moreira, Basilio, & Lafortune, 2013). Differently from the works by Alves et al. (2014), Carvalho et al. (2012, 2013), Rohloff (2005) and Sanchez and Montoya (2006) we are not proposing a system that copes with sensor faults but one that actually detects its malfunction.

This paper is organized as follows. We present in Section 2 a brief review of Discrete Event Systems (DES) theory and review the model for intermittent loss of observations proposed in Carvalho et al. (2012). In Section 3, we convert the problem of sensor fault diagnosis into an equivalent one that consists of diagnosing the language generated by an automaton subject to intermittent sensor faults, where the fault event is the event recorded by the sensor whose malfunction must be diagnosed, and present the definitions of *F*-, *R*-, and *FR*-diagnosability. After that, we present necessary and sufficient conditions for the diagnosis of intermittent faults using diagnosers (Section 4) and verifiers (Section 5). Finally, in Section 6, we remind the main contributions of the paper.

2. Preliminaries

Let $G = (X, \Sigma, f, \Gamma, x_0, X_m)$ be a deterministic automaton, where X denotes the state space, Σ is the finite set of events, $f : X \times \Sigma \to X$ is the state transition function, $\Gamma : X \to 2^{\Sigma}$ is the active event function, where $\Gamma(x) = \{\sigma \in \Sigma : f(x, \sigma) \text{ is defined}\}$, x_0 is the initial state, and X_m is the set of marked states. When the set of marked states is empty, *i.e.*, $X_m = \emptyset$, it will be omitted from G. The Kleene-closure of Σ, Σ^*, is the set of all possible finite length traces that can be formed with the elements of Σ, including the empty trace ϵ. We extend the domain of f to $X \times \Sigma^*$ to define the language generated by G (denoted as $L(G)$, or simply, L) as the set of all traces $s \in \Sigma^*$ for which $f(x_0, s)$ is defined.

The accessible part of G, denoted by $Ac(G)$, is the unary operation that deletes from G the states that are not reachable from x_0 and the transitions attached to these states, *i.e.*, $Ac(G) = (X_{ac}, \Sigma, f_{ac}, \Gamma_{ac}, x_0, X_{ac,m})$, where $X_{ac} = \{x \in X : (\exists s \in \Sigma^*)[f(x_0, s) = x]\}$, $f_{ac} : X_{ac} \times \Sigma \to X_{ac}$, $\Gamma_{ac} : X_{ac} \to 2^{\Sigma}$, and $X_{ac,m} = X_m \cap X_{ac}$. The coaccessible part of G, denoted as $CoAc(G)$, is obtained by deleting all states of G from which it is not possible to reach a marked state and their associated transitions, *i.e.*, $CoAc(G) = (X_{coac}, \Sigma, f_{coac}, \Gamma_{coac}, x_{0,coac}, X_m)$ where $X_{coac} = \{x \in X : (\exists s \in \Sigma^*)[f(x, s) \in X_m]\}$, $f_{coac} : X_{coac} \times \Sigma \to X_{coac}$, with $f_{coac}(x, \sigma) = f(x, \sigma)$, if $x \in X_{coac}$ and $f(x, \sigma) \in X_{coac}$, or undefined, otherwise, and $\Gamma_{coac} : X_{coac} \to 2^{\Sigma}$, with $\Gamma_{coac}(x_{coac}) = \{\sigma : \sigma \in \Sigma, f_{coac}(x, \sigma) \text{ is defined}\}$, and $x_{0,coac} = x_0$, if $x_0 \in X_{coac}$, or undefined, if $x_0 \notin X_{coac}$.

Let $G_1 = (X_1, \Sigma_1, f_1, \Gamma_1, x_{0,1})$ and $G_2 = (X_2, \Sigma_2, f_2, \Gamma_2, x_{0,2})$ denote two finite state automata. The parallel composition between G_1 and G_2 (denoted as $G_1 \parallel G_2$) is defined as $G_1 \parallel G_2 = (X_1 \times X_2, \Sigma_1 \cup \Sigma_2, f_{1\parallel 2}, \Gamma_{1\parallel 2}, (x_{0,1}, x_{0,2}))$, where $f_{1\parallel 2} : (X_1 \times X_2) \times (\Sigma_1 \cup \Sigma_2) \to (X_1 \times X_2)$ is defined as follows: $f_{1\parallel 2}((x_1, x_2), \sigma) = (f_1(x_1, \sigma), x_2)$ if $\sigma \in \Gamma_1(x_1) \setminus \Sigma_2$; $f_{1\parallel 2}((x_1, x_2), \sigma) = (x_1, f_2(x_2, \sigma))$ if $\sigma \in \Gamma_2(x_2) \setminus \Sigma_1$; $f_{1\parallel 2}((x_1, x_2), \sigma) = (f_1(x_1, \sigma), f_2(x_2, \sigma))$ if $\sigma \in \Gamma_1(x_1) \cap \Gamma_2(x_2)$; and undefined, otherwise; and for all $(x_1, x_2) \in X_1 \times X_2$, $\sigma \in \Sigma_1 \cup \Sigma_2$, $\Gamma_{1\parallel 2}((x_1, x_2)) = (\Gamma_1(x_1) \cap \Gamma_2(x_2)) \cup (\Gamma_1(x_1) \setminus \Sigma_2) \cup (\Gamma_2(x_2) \setminus \Sigma_1)$.

Let $\Sigma = \Sigma_o \dot{\cup} \Sigma_{uo}$ be a partition of Σ, where Σ_o and Σ_{uo} are, respectively, the set of observable and unobservable events. An important language operation is the natural projection $P_o : \Sigma^* \to \Sigma_o^*$ satisfying the following properties (Ramadge & Wonham, 1989): (i) $P_o(\epsilon) = \epsilon$, (ii) $P_o(\sigma) = \sigma$, if $\sigma \in \Sigma_o$, or $P_o(\sigma) = \epsilon$, if $\sigma \in \Sigma_{uo}$ and, $P_o(s\sigma) = P_o(s)P_o(\sigma), s \in \Sigma^*, \sigma \in \Sigma$. The projection operation can be extended to a language L by applying the natural projection to all traces of L. Therefore, if $L \subseteq \Sigma^*$, then $P_o(L) = \{t \in \Sigma_o^* : (\exists s \in L)[P_o(s) = t]\}$. The inverse projection P_o^{-1} is defined as $P_o^{-1}(s) = \{t \in \Sigma^* : P_o(t) = s\}$.

The observed dynamic behavior of a deterministic automaton G with unobservable events, can be described by a deterministic automaton called observer (denoted as Obs (G)), whose event set is the set of observable events of G and the states are estimates of the states of the plant G after the observation of a trace. The language generated by Obs (G) is the projection of the language generated by G over Σ_o^*, *i.e.*, $L(\text{Obs}\,(G)) = P_o[L(G)]$ (Cassandras & Lafortune, 2008).

Let $\Sigma_{isf} \subseteq \Sigma_o$ denote the set of events associated with the sensors that are subject to intermittent faults, and define $\Sigma'_{isf} = \{\sigma' : \sigma \in \Sigma_{isf}\}$ and $\Sigma_{dil} = \Sigma \dot{\cup} \Sigma'_{isf}$. The following language operation can be defined (Carvalho et al., 2012).

Definition 1 (*Dilation*)**.** The dilation D is the mapping $D : \Sigma^* \to 2^{\Sigma_{dil}^*}$, where $D(\epsilon) = \{\epsilon\}$, $D(\sigma) = \{\sigma\}$, if $\sigma \in \Sigma \setminus \Sigma_{isf}$, $D(\sigma) = \{\sigma, \sigma'\}$, if $\sigma \in \Sigma_{isf}$, and $D(s\sigma) = D(s)D(\sigma)$, $s \in \Sigma^*, \sigma \in \Sigma$.