Automatica 105 (2019) 376–383

Contents lists available at ScienceDirect

Automatica

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

Brief paper

A general approach for optimizing dynamic sensor activation for discrete event systems[☆]

Xiang Yin[a,b,*], Stéphane Lafortune[c]

[a] Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China
[b] Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai 200240, China
[c] Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA

ARTICLE INFO

Article history:
Received 29 June 2017
Received in revised form 12 June 2018
Accepted 7 February 2019
Available online xxxx

Keywords:
Discrete event systems
Sensor activation
Property enforcement
Optimization

ABSTRACT

We study the problem of dynamic sensor activation for centralized partially-observed discrete event systems. The sensors can be turned on/off online dynamically according to a sensor activation policy in order to satisfy some observation property. We consider a general class of properties, called Information-State-based (or IS-based) properties, which include, but are not limited to, observability, K-diagnosability, predictability, and opacity. We define a new Most Permissive Observer (MPO) that generalizes previous versions of this structure. Based on the generalized MPO, we first synthesize a logical minimal or maximal sensor activation policy based on a set inclusion criterion. Then we study the synthesis of optimal solutions for a given quantitative objective function that considers numerical activation costs and switching costs. Our results generalize previous works on dynamic sensor activation for enforcement of specific properties.

1. Introduction

We consider dynamic sensor activation problem in centralized and partially-observed Discrete Event Systems (DES). The objective in this problem is to synthesize a sensor activation policy that dynamically turns sensors on/off online in order to achieve a given objective, e.g., to control the system or to diagnose faults. This problem is important since in many applications turning more sensors on implies that more energy or bandwidth is consumed. Therefore, it is of interest to synthesize a sensor activation policy that is optimal with respect to some criterion, subject to the constraints of the problem.

Dynamic sensor activation has been studied extensively in the DES literature; see, e.g., Cassez, Dubreil, and Marchand (2012), Cassez and Tripakis (2008), Dallal and Lafortune (2014), Shu, Huang, and Lin (2013), Thorsley and Teneketzis (2007), Wang, Lafortune, Lin and Girard (2010), Wang, Lafortune, Girard and Lin (2010), Yin and Lafortune (2018), Zhang, Shu, and Lin (2015) and the recent survey paper (Sears & Rudie, 2016) for an extensive bibliography. In Cassez and Tripakis (2008) and Thorsley and Teneketzis (2007), the problem of dynamic sensor activation for the purpose of fault diagnosis was studied. In Wang, Lafortune and Girard et al. (2010) and Wang, Lafortune and Lin et al. (2010), both centralized and decentralized sensor activation problems for the purposes of control and diagnosis, respectively, were studied. In Shu et al. (2013), an online approach was proposed for detectability.

In Cassez and Tripakis (2008), a game structure called the *Most Permissive Observer* (MPO) was proposed for solving the problem of dynamic sensor activation for the purpose of fault diagnosis. The MPO is a finite structure that embeds, in some sense to be made precise later in this paper, all valid sensor activation policies, i.e., all policies that enforce the property of K-diagnosability. This approach was extended to timed systems in Cassez (2010) and to the problem of opacity in Cassez et al. (2012). An information-state-based characterization of the MPO structure was proposed in Dallal and Lafortune (2014) in the context of the enforcement of K-diagnosability. Similar game-theoretical approach has also been used in Wu and Lafortune (2014) and Yin and Lafortune (2016) for different purposes.

In this paper, we use the MPO approach to investigate the sensor activation problem for centralized partially-observed DES. However, instead of investigating the enforcement of a particular property, e.g., observability, diagnosability, or opacity, as

[☆] This work is supported by the National Natural Science Foundation of China (61803259, 61833012), NSF grants CCF-1138860 (Expeditions in Computing project ExCAPE: Expeditions in Computer Augmented Program Engineering), CNS-1421122, and CNS-1446298. The material in this paper was partially presented at the 54th IEEE Conference on Decision and Control, December 15–18, 2015, Osaka, Japan. This paper was recommended for publication in revised form by Associate Editor Joerg Raisch under the direction of Editor Christos G. Cassandras.

[*] Corresponding author at: Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China.

E-mail addresses: yinxiang@sjtu.edu.cn (X. Yin), stephane@umich.edu (S. Lafortune).

https://doi.org/10.1016/j.automatica.2019.04.001
0005-1098/© 2019 Elsevier Ltd. All rights reserved.

Table 1
Comparison between the proposed general approach and previous approaches in sensor activation problems.

	Cost	Property					
		Observability	Diagnosability	Detectability	Predictability	Opacity	Anonymity
Previous results	Logical cost	Wang, Lafortune and Lin et al. (2010)	Dallal and Lafortune (2014), Wang, Lafortune and Girard et al. (2010)	Shu et al. (2013)	N/A	Zhang et al. (2015)	N/A
	Activation cost	N/A	Cassez and Tripakis (2008), Thorsley and Teneketzis (2007)	N/A	N/A	Cassez et al. (2012)	N/A
	Switching cost	N/A	N/A	N/A	N/A	N/A	N/A
This paper		All three costs for all the above properties					

was done in previous works, we study a *general class of properties* called Information-State-based (IS-based) properties, that captures all properties previously considered, and more. Specifically, we formulate the problem of dynamic sensor activation for any property that can be expressed as an IS-based property. To solve this problem, we define a generalized version of the most permissive observer. Then we present algorithms for the synthesis of optimal sensor activation policies under qualitative or quantitative performance objectives.

Compared with prior works where the MPO was employed (Cassez et al., 2012; Cassez & Tripakis, 2008; Dallal & Lafortune, 2014), the MPO defined in this paper is more general since we consider a general class of properties. The problem of optimal sensor activation for predictability, which to the best of our knowledge, has not been considered so far in the literature, can also be solved by our approach. Moreover, our approach can be employed to solve sensor activation problems for the enforcement of a wide class of user-defined properties that can be expressed as IS-based properties. Also, we solve a quantitative optimization problem by considering numerical *activation costs* and *switching costs*. Previously, only activation costs were considered in the literature (Cassez et al., 2012; Cassez & Tripakis, 2008; Thorsley & Teneketzis, 2007). However, considering switching costs is important in many applications, since turning a sensor on/off too frequently may decrease its life span. To the best of our knowledge, such a switching cost for sensor activation has never been considered in the DES literature. Our proposed approach is compared with previous works in Table 1, where all of the problems listed can be solved with our new approach. Preliminary and partial versions of some of the results in Sections 3, 4, and 6 appear in Yin and Lafortune (2015).

2. Preliminaries and problem formulation

2.1. System model

The system under consideration is modeled by a deterministic finite state automaton $G = (Q, \Sigma, \delta, q_0)$, where Q is the finite set of states, Σ is the finite set of events, $\delta : Q \times \Sigma \to Q$ is the partial transition function, and q_0 is the initial state. The transition function δ is extended to $Q \times \Sigma^*$ in the usual manner (see, e.g., Cassandras & Lafortune, 2008). The language generated by G from state q is defined by $\mathcal{L}(G, q) = \{s \in \Sigma^* : \delta(q, s)!\}$, where ! means "is defined". We write $\mathcal{L}(G, q)$ as $\mathcal{L}(G)$ if $q = q_0$. The prefix-closure of a language L is $\overline{L} = \{s \in \Sigma^* : \exists w \in \Sigma^* \text{ s.t. } sw \in L\}$. We use notation $|\cdot|$ to denote the length of a string. We denote by L/s the post-language of L after s, i.e., $L/s = \{t \in \Sigma^* : st \in L\}$. We say a language L is live if $\forall s \in L, \exists \sigma \in \Sigma : s\sigma \in L$. Hereafter, we assume w.l.o.g. that $\mathcal{L}(G)$ is live.

The sensors are turned on/off dynamically based on the observation history. When the sensor corresponding to an event $\sigma \in \Sigma$ is turned "on", we say that the event is being *monitored*. While an event is monitored, any occurrence of it will be *observed*. The set of events $\theta \in 2^\Sigma$ that we decide to monitor, at any point, is called a *sensing decision*. We assume that Σ is partitioned as $\Sigma = \Sigma_o \dot{\cup} \Sigma_s \dot{\cup} \Sigma_{uo}$, where: (i) Σ_o is the set of events whose occurrences are always observed; (ii) Σ_s is the set of events that we can choose to monitor or not; and (iii) Σ_{uo} is the set of events that are always unobservable. We say that a sensing decision $\theta \in 2^\Sigma$ is *admissible* if $\Sigma_o \subseteq \theta \subseteq \Sigma_o \cup \Sigma_s$ and we let Θ denote the set of all admissible sensing decisions.

Under dynamic sensing decisions, the observations of the system behavior are specified by a *sensor activation policy* $\omega : \mathcal{L}(G) \to \Theta$, where for any $s \in \mathcal{L}(G)$, $\omega(s)$ is the set of events that are monitored after the occurrence of s. Given a sensor activation policy ω, we define the projection $P_\omega : \mathcal{L}(G) \to \Sigma^*$ recursively by:

$$P_\omega(\epsilon) = \epsilon, \quad P_\omega(s\sigma) = \begin{cases} P_\omega(s)\sigma & \text{if } \sigma \in \omega(s) \\ P_\omega(s) & \text{if } \sigma \notin \omega(s) \end{cases} \tag{1}$$

We also define the *state estimator function* (or simply "state estimator") under ω, $\mathcal{E}^G_\omega : \mathcal{L}(G) \to 2^Q$, as follows upon the occurrence of $s \in \mathcal{L}(G)$:

$$\mathcal{E}^G_\omega(s) := \{q \in Q : \exists t \in \mathcal{L}(G) \text{ s.t. } P_\omega(s) = P_\omega(t) \wedge \delta(q_0, t) = q\} \tag{2}$$

For the purpose of implementation, we require that $\forall s, t \in \mathcal{L}(G) : P_\omega(s) = P_\omega(t) \Rightarrow \omega(s) = \omega(t)$. It simply requires that the sensing decisions for any two indistinguishable strings must be the same. We use the notation Ω to denote the set of all sensor activation policies. Given two sensor activation policies $\omega, \omega' \in \Omega$, we say that ω is *smaller* than ω', denoted by $\omega < \omega'$, if (1) $\forall s \in \mathcal{L}(G) : \omega(s) \subseteq \omega'(s)$; and (2) $\exists s \in \mathcal{L}(G) : \omega(s) \subset \omega'(s)$.

2.2. Problem formulation

We define an *information state* to be a subset of states in Q and denote by $I = 2^Q$ the set of information states. Hereafter, we consider a special class of properties called *information-state-based (IS-based) properties.*

Definition 1. Let G be the system automaton and $\omega : \mathcal{L}(G) \to \Theta$ be a sensor activation policy. An IS-based property w.r.t. G is a function $\varphi : 2^Q \to \{0, 1\}$. We say that ω satisfies φ w.r.t. G, denoted by $\omega \models_G \varphi$, if $\forall s \in \mathcal{L}(G) : \varphi(\mathcal{E}^G_\omega(s)) = 1$.

Example 1. Consider the system G in Fig. 1. Let $\varphi : 2^Q \to \{0, 1\}$ be an IS-based property defined by: for any $i \in 2^Q$, $\varphi(i) = 1$ if and only if $\nexists q \in \{1, 4, 5, 6\} : \{3, q\} \subseteq i$. This IS-based property φ requires that we should never confuse state 3 with any state in $\{1, 4, 5, 6\}$. Let us consider the sensor activation policy ω defined by $\forall s \in \mathcal{L}(G) : \omega(s) = \{o\}$. By taking $eo \in \mathcal{L}(G)$, we know that $\mathcal{E}^G_\omega(eo) = \{1, 2, 3, 4, 5, 6, 7\}$. Therefore, $\omega \not\models_G \varphi$.

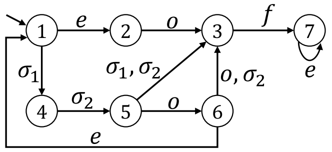


Fig. 1. System with $\Sigma_o=\{o\}$, $\Sigma_s=\{\sigma_1,\sigma_2\}$ and $\Sigma_{uo}=\{e,f\}$.

Our objective is to synthesize a sensor activation policy such that some given property holds. We define the *Minimal Sensor Activation Problem for IS-Based Properties* as follows.

Problem 1. Let G be the system automaton and φ be an IS-based property w.r.t. G. Find a sensor activation policy $\omega\in\Omega$ s.t. (i) $\omega\models_G\varphi$; and (ii) $\nexists\omega'\in\Omega$ s.t. $\omega'\models_G\varphi$ and $\omega'<\omega$.

In some contexts, we may be interested in the dual version of the Minimal Sensor Activation Problem, the *Maximal Sensor Activation Problem for IS-Based Properties*. Its definition is analogous, with "$<$" replaced by "$>$" in condition (ii).

Remark 1. In Wang, Lafortune, and Lin (2007), the *state disambiguation problem* is defined. Formally, $T_{spec}\subseteq Q\times Q$ is the set of state pairs that need to be distinguished and we want to find a minimal $\omega\in\Omega$ s.t. $(\forall s\in\mathcal{L}(G))(\forall q_1,q_2\in\mathcal{E}^G_\omega(s))[(q_1,q_2)\notin T_{spec}]$. This problem is a special case of the minimal sensor activation problem for IS-based properties, since given T_{spec}, we can define an IS-based property $\varphi_{spec}:2^Q\to\{0,1\}$ by: $\forall i\in 2^Q:[\varphi_{spec}(i)=0]\Leftrightarrow[\exists q_1,q_2\in i:(q_1,q_2)\in T_{spec}]$.

Remark 2. In some scenarios, e.g., when the system is monitored by a malicious *external* observer, the "disablement" of sensors can be costly, since we need to spend additional effort to hide the occurrences of the corresponding events. In this regard, the optimal dynamic mask synthesis problem investigated in the literature (see, e.g., Cassez et al., 2012) is essentially the maximal sensor activation problem defined above.

Remark 3. In Yin and Lafortune (2016), a similar concept of IS-based properties was defined for the purpose of *control*. However, the notion of IS-based property defined in this paper and the related notion of IS-based property in the control problem are incomparable. For example, in the supervisory control problem, safety is an IS-based property but cannot be formulated as an IS-based property in the sensor activation problem. On the other hand, there exist some properties that are IS-based properties in the sensor activation problem, but that cannot be formulated as IS-based properties in the control problem. One such example is predictability; we will further elaborate on this issue in Section 6.2.

3. A general most permissive observer

3.1. Information state dynamics

A sensor activation policy ω works dynamically as follows. Initially, a sensing decision θ_0 is issued. Then, upon the occurrence of (monitored) event $\sigma_1\in\theta_0$, a new decision θ_1 is made, and so forth. We call such a sequence in the form of $\theta_0\sigma_1\theta_1\sigma_2\ldots$, where $\theta_i\in\Theta,\sigma_{i+1}\in\theta_i,\forall i\geq 0$, a *run*. For any $s\in\mathcal{L}(G)$, suppose that $s=\xi_0\sigma_1\xi_1\sigma_2\ldots\xi_{n-1}\sigma_n\xi_n$, where $\xi_i\in(\Sigma\setminus\omega(\xi_0\sigma_1\ldots\xi_{i-1}\sigma_i))^*,\forall i\geq 0$ and $\sigma_i\in\omega(\xi_0\sigma_1\ldots\sigma_{i-1}\xi_{i-1}),\forall i\geq 1$. In words, ξ_i is an unobserved string and σ_i is a monitored event. Then the information available to the sensor activation module upon the occurrence of s is, in fact, the run

$$\mathcal{R}_\omega(s):=\theta_0\sigma_1\theta_1\ldots\theta_{n-1}\sigma_n\theta_n \tag{3}$$

where $\theta_i=\omega(\xi_0\sigma_1\ldots\xi_{i-1}\sigma_i\xi_i),\forall i\geq 0$.

To capture the alternating nature of sensing decisions and observations, we define two kinds of states, termed Y-states and Z-states, respectively. A Y-state y is an information state from which a sensing decision is made and $Y\subseteq I$ denotes the set of Y-states. A Z-state z is an information state augmented with a sensing decision from which observations of monitored events occur. $Z\subseteq I\times\Theta$ denotes the set of Z-states and we write $z=(I(z),\Theta(z))$ for any $z\in Z$. We define the transition function from Y-states to Z-states, $h_{YZ}:Y\times\Theta\to Z$, and the transition function from Z-states to Y-states, $h_{ZY}:Z\times\Sigma\to Y$. For any $y\in I,z\in I\times\Theta,\sigma\in\Sigma$ and $\theta\in\Theta$,

- $z=h_{YZ}(y,\theta)$ if and only if $I(z)=\{q\in Q:\exists q'\in y,s\in(\Sigma\setminus\theta)^*\text{ s.t. }\delta(q',s)=q\}$ and $\Theta(z)=\theta$
- $y=h_{ZY}(z,\sigma)$ if and only if $\sigma\in\Theta(z)$ and $y=\{q\in Q:\exists q'\in I(z)\text{ s.t. }\delta(q',\sigma)=q\}$

For simplicity hereafter, we write $y\xrightarrow{\theta}z$ if $z=h_{YZ}(y,\theta)$ and $z\xrightarrow{\sigma}y$ if $z=h_{ZY}(z,\sigma)$. Intuitively, $y\xrightarrow{\theta}z$ simply represents the *unobserved reach* under sensing decision θ and it remembers the sensing decision that leads to it. On the other hand, $z\xrightarrow{\sigma}y$ represents the set of states the system can reach *immediately after* the occurrence of event σ. We require that $\sigma\in\Theta(z)$, since σ must be monitored.

Let $s\in\mathcal{L}(G)$ be a string and $\mathcal{R}_\omega(s)=\theta_0\sigma_1\theta_1\ldots\theta_{n-1}\sigma_n\theta_n$ be its corresponding run defined in Eq. (3). Let $y_0=\{q_0\}$ be the initial Y-state. Then occurrence of the run $\theta_0\sigma_1\theta_1\ldots\theta_{n-1}\sigma_n\theta_n$ will reach an alternating sequence of Y- and Z-states

$$y_0\xrightarrow{\theta_0}z_0\xrightarrow{\sigma_1}y_1\xrightarrow{\theta_1}\ldots\xrightarrow{\theta_{n-1}}z_{n-1}\xrightarrow{\sigma_n}y_n\xrightarrow{\theta_n}z_n \tag{4}$$

We denote by $\mathcal{I}^Y_\omega(s)$ and $\mathcal{I}^Z_\omega(s)$, the last Y-state and Z-state in $y_0z_0y_1z_2\ldots z_{n-1}y_nz_n$, respectively, i.e., $\mathcal{I}^Y_\omega(s)=y_n$ and $\mathcal{I}^Z_\omega(s)=z_n$. By induction on the length of $P_\omega(s)$, it can be verified that $I(\mathcal{I}^Z_\omega(s))=\mathcal{E}^G_\omega(s)$, i.e., the information state component of $\mathcal{I}^Z_\omega(s)$ is the state estimator of s.

Example 2. Let us return to the system G in Fig. 1. Consider the sensor activation policy ω defined by:

$$\omega(s)=\begin{cases}\{o,\sigma_1\}, & \text{if } s\in\{\epsilon,e\}\\ \{o\}, & \text{otherwise}\end{cases} \tag{5}$$

The above definition means that ω monitors event σ_1 only when nothing has been observed so far. Let us consider the string $s=\sigma_1\sigma_2$. The corresponding run of s is $\mathcal{R}_\omega(\sigma_1\sigma_2)=\{o,\sigma_1\}\sigma_1\{o\}$ and the corresponding sequence of Y- and Z-states is $\{1\}\xrightarrow{\{o,\sigma_1\}}(\{1,2\},\{o,\sigma_1\})\xrightarrow{\sigma_1}\{4\}\xrightarrow{\{o\}}(\{3,4,5,7\},\{o\})$. So we have that $\mathcal{I}^Y_\omega(\sigma_1\sigma_2)=\{4\}$, $\mathcal{I}^Z_\omega(\sigma_1\sigma_2)=(\{3,4,5,7\},\{o\})$ and $\mathcal{E}^G_\omega(\sigma_1\sigma_2)=I(\mathcal{I}^Z_\omega(\sigma_1\sigma_2))=\{3,4,5,7\}$.

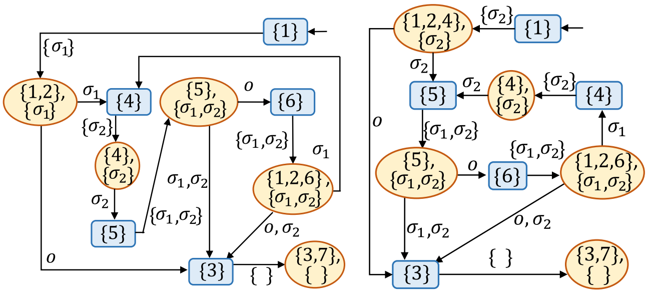
3.2. Bipartite dynamic observer

Recall that the sensor activation policy ω is a function defined over a language domain. For implementation purposes, we need to build a *finite representation* of the function ω. To this end, we define the structure of bipartite dynamic observer (BDO) that realizes a (set of) sensor activation policy(ies).

Definition 2. A bipartite dynamic observer $\mathcal{O}$ is a 7-tuple

$$\mathcal{O}=(Q^{\mathcal{O}}_Y,Q^{\mathcal{O}}_Z,h^{\mathcal{O}}_{YZ},h^{\mathcal{O}}_{ZY},\Sigma,\Theta,y_0) \tag{6}$$

where, $Q^{\mathcal{O}}_Y\subseteq I$ is a set of Y-states, $Q^{\mathcal{O}}_Z\subseteq I\times\Theta$ is a set of Z-states, $h^{\mathcal{O}}_{YZ}:Q^{\mathcal{O}}_Y\times\Theta\to Q^{\mathcal{O}}_Z$ and $h^{\mathcal{O}}_{ZY}:Q^{\mathcal{O}}_Z\times\Sigma\to Q^{\mathcal{O}}_Y$ are partial transition functions such that for any $z\in Q^{\mathcal{O}}_Z,y\in Q^{\mathcal{O}}_Y,\theta\in\Theta$ and $\sigma\in\Sigma$, the following conditions hold



(a) Solution $\mathcal{O}_1$ (b) Solution $\mathcal{O}_2$

Fig. 2. Two incomparable minimal solutions.

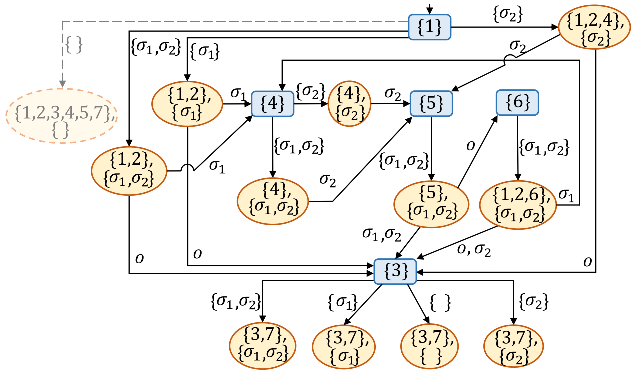


Fig. 3. Example of BDOs, where blue rectangular states and yellow oval states represent, respectively, Y-states and Z-states.

C1. $h^{\mathcal{O}}_{ZY}(z,\sigma)=y \Leftrightarrow h_{ZY}(z,\sigma)=y$;
C2. $h^{\mathcal{O}}_{YZ}(y,\theta)=z \Rightarrow h_{YZ}(y,\theta)=z$;
C3. $\forall y\in Q^{\mathcal{O}}_Y, \exists \theta\in\Theta : h^{\mathcal{O}}_{YZ}(y,\theta)!$.

Σ is the set of events of G, Θ is the set of admissible sensing decisions, and $y_0=\{q_0\}$ is the initial Y-state. For brevity, we only consider the accessible part of a BDO.

Condition C1 says that for any $z\in Q^{\mathcal{O}}_Z$, $h^{\mathcal{O}}_{ZY}(z,\sigma)$ is defined for any possible observation $\sigma\in\Theta(z)$ by the definition of h_{ZY}. This is due to the fact that we cannot decide which monitored event will occur once we make a sensing decision. Condition C2 says that for the transition function $h^{\mathcal{O}}_{YZ}$, we have either $h^{\mathcal{O}}_{YZ}(y,\theta)=h_{YZ}(y,\theta)$ or it is undefined. Condition C3 requires that for any Y-state $y\in Q^{\mathcal{O}}_Y$, there exists at least one $\theta\in\Theta$ such that $h^{\mathcal{O}}_{YZ}(y,\theta)$ is defined. This is because a sensor activation policy is defined for all strings in $\mathcal{L}(G)$ and we must make a sensing decision at all accessible Y-states.

Definition 3. Given a BDO $\mathcal{O}$, we say that ω is *allowed* by $\mathcal{O}$ if $\forall s\in\mathcal{L}(G) : h^{\mathcal{O}}_{YZ}(\mathcal{I}^Y_\omega(s),\omega(s))!$. With a slight abuse of notation, we write that $\omega\in\mathcal{O}$ whenever ω is allowed by $\mathcal{O}$.

Given a BDO $\mathcal{O}$, the set of sensor activation policies allowed by $\mathcal{O}$ may not be a singleton, since for each Y-state there may be multiple sensing decisions to choose from. Moreover, the domain of a sensor activation policy in a BDO need not be finite since different sensing decisions may be chosen on different visits to the same Y-state. We say that a BDO $\mathcal{O}$ is *deterministic* if, for any $y\in Q^{\mathcal{O}}_Y$, there exists only one $\theta\in\Theta$ such that $h^{\mathcal{O}}_{YZ}(y,\theta)!$. It is clear that a deterministic BDO $\mathcal{O}$ allows a unique sensor activation policy; we denote it by $\omega_{\mathcal{O}}$. In this case, the deterministic BDO $\mathcal{O}$ is essentially a *finite representation* of $\omega_{\mathcal{O}}$.

Example 3. Consider again the system G in Fig. 1. Fig. 2(a) provides an example of a deterministic BDO. For the sake of simplicity, since event $o\in\Sigma_o$ is always observable, we omitted this event in each sensing decision in the figures. For the initial Y-state $y_0=\{1\}$, by making sensing decision $\theta=\{o,\sigma_1\}$, we will reach Z-state $z=h_{YZ}(y_0,\theta)=(\{1,2\},\{o,\sigma_1\})$. From z, only monitored events o and σ_1 can be observed. If σ_1 is observed, then the next Y-state is $y_1=h_{ZY}(z,\sigma_1)=\{4\}$, and so forth. Similarly, the BDO $\mathcal{O}_2$ shown in Fig. 2(b) is also deterministic. However, the BDO shown in Fig. 3 is not a deterministic BDO, since there are three sensing decisions $\{o,\sigma_1\}$, $\{o,\sigma_2\}$ and $\{o,\sigma_1,\sigma_2\}$ defined at Y-state $\{1\}$.

3.3. Generalized MPO and its properties

We return to the sensor action problem for IS-based properties, Problem 1, formulated in Section 2.2. By condition (i) in Problem 1, we must find an ω such that $\forall s\in\mathcal{L}(G):\varphi(\mathcal{E}^G_\omega(s))=1$. However, for any BDO, we know that $\forall s\in\mathcal{L}(G): I(\mathcal{I}^Z_\omega(s))=\mathcal{E}^G_\omega(s)$ and $\mathcal{I}^Z_\omega(s)$ is indeed the Z-state reached by the run $\mathcal{R}_\omega(s)$ in the BDO. Therefore, if we construct a BDO $\mathcal{O}$ such that

$$\forall z\in Q^{\mathcal{O}}_Z : \varphi(I(z))=1 \tag{7}$$

and such that $\mathcal{O}$ is "as large as possible", then the resulting structure will contain all sensor activation policies that satisfy φ. The property of such a BDO being as large as possible is actually well defined: if $\mathcal{O}_1$ and $\mathcal{O}_2$ are two BDOs that both satisfy Eq. (7), then their union, in the sense of graph merger, is a BDO that satisfies Eq. (7). This observation leads to the definition of the most permissive observer.

Definition 4 (*Most Permissive Observer*)**.** Let G be the system and let φ be the IS-based property under consideration. The Most Permissive Observer for φ is the BDO

$$\mathcal{MPO}_\varphi=(Q^{MPO}_Y, Q^{MPO}_Z, h^{MPO}_{YZ}, h^{MPO}_{ZY}, \Sigma, \Theta, y_0)$$

defined as the largest BDO such that $\forall z\in Q^{MPO}_Z : \varphi(I(z))=1$.

Note that, in our definitions, the BDO is a class of bipartite structures, while the MPO is a specific type of BDO. The following theorem reveals that the MPO embeds all sensor activation policies satisfying φ in its structure. Due to space constraints, its proof has been omitted and it is available in Yin and Lafortune (0000).

Theorem 1. *$\omega\models_G\varphi$ if and only if $\omega\in\mathcal{MPO}_\varphi$.*

The MPO can be constructed directly based on its definition. First, we search through the state space of Y-states and Z-states until a Z-state that violates the IS-based property φ is encountered. Then we iteratively remove Y-state that has no successors and Z-states that has undefined feasible outgoing transitions, until the structure converges to a BDO. The worst-case time complexity of the construction of the MPO is exponential in both $|Q|$ and $|\Sigma_s|$. In Yin and Lafortune (0000), a detailed algorithm is provided for the construction of the MPO; the construction algorithm has also been implemented by software DPO-SYNT; see, https://github.com/xiang-yin/DPO-SYNT.

Example 4. We return to system G in Fig. 1 and IS-based property φ defined in Example 1. The corresponding MPO is shown in Fig. 3. At initial Y-state $\{1\}$, if we make sensing decision $\{o\}$ (depicted as $\{\}$ in the figure), then Z-state $\{1,2,3,4,5,7\}$ will be reached (see the dashed lines). However, the information-state component of this Z-state contains both states 3 and 1, i.e., the IS-based property φ is violated. Therefore, we cannot make sensing decision $\{o\}$ at the initial state.

4. Synthesis of logically optimal policies

In this section, we show how to synthesize from the MPO an optimal sensor activation policy ω that solves Problem 1. Particularly, we shall require that ω be defined over a finite domain, so that it can be effectively implemented. To this end, we define a special class of sensor activation policies that are represented by subgraphs of the MPO and thus have finite realizations.

Definition 5. A sensor activation policy ω is information-state-based if $\forall s, t \in \mathcal{L}(G) : \mathcal{I}^Y_\omega(s) = \mathcal{I}^Y_\omega(t) \Rightarrow \omega(s) = \omega(t)$.

Clearly, if ω is IS-based, then ω can always be represented by a deterministic BDO that is a subgraph of the MPO.

Definition 6. Suppose ω is a sensor activation policy such that $\omega \models_G \varphi$. We say that ω is greedy minimal if $\forall s \in \mathcal{L}(G), \forall \theta \in \Theta : h^{MPO}_{YZ}(\mathcal{I}^Y_\omega(s), \theta)! \Rightarrow \theta \not\subset \omega(s)$. The notion of greedy maximality is defined analogously.

The following theorem says that a greedy minimal (respectively, maximal) solution is a minimal (respectively, maximal) solution. Its proof is available in Yin and Lafortune (0000).

Theorem 2. *Suppose ω is a sensor activation policy such that $\omega \models_G \varphi$. Then ω is minimal (respectively, maximal) if it is greedy minimal (respectively, greedy maximal).*

By Theorem 2, it is clear that if we synthesize an IS-based greedy optimal sensor activation policy, then we will have obtained a solution to Problem 1, which was our objective. (Of course, not all solutions to Problem 1 need be IS-based or greedy.) An IS-based greedy optimal sensor activation policy can be obtained by a depth-first search over the state space of the MPO that picks *one* greedy optimal sensing decision at each Y-state and then picks *all* observations for each Z-state. The resulting structure will be a deterministic BDO that represents the solution. We illustrate this synthesis procedure by an example.

Example 5. We return to the MPO shown in Fig. 3. To synthesize a minimal sensor activation policy for φ, we can pick decision $\{o, \sigma_1\}$, which is greedy minimal, at the initial Y-state. Then, upon the occurrence of monitored event σ_1, the new Y-state $\{4\}$ is reached. At that state, we pick the unique greedy minimal decision $\{o, \sigma_2\}$, and so forth. These choices result in deterministic BDO $\mathcal{O}_1$ shown in Fig. 2(a) that allows the unique sensor activation policy $\omega_{\mathcal{O}_1}$, which is provably minimal.

Remark 4. In the synthesis step in the previous example, we could have selected sensing decision $\{o, \sigma_2\}$ at the initial Y-state, which yields the minimal solution shown in Fig. 2(b). Interestingly, we see that the intersection of the two valid decisions $\{o, \sigma_1\}$ and $\{o, \sigma_2\}$, i.e., $\{o\}$ is not a valid decision, since $\{o\}$ is not defined at Y-state $\{1\}$ in the MPO (recall the discussion in Example 4). This illustrates the earlier claim that Problem 1 may not have an infimal (respectively, supremal) solution in general, but instead several incomparable minimal (respectively maximal) solutions.

5. Synthesis of numerically optimal policies

In Section 4, we solved the optimal sensor activation problem under a logical performance objective based on set inclusion. However, in many applications, it may be useful or preferable to quantify the performance of a sensor activation policy under a *numerical* performance objective. The problem of quantitative optimization is addressed in this section.

We consider two different types of numerical costs: activation costs and switching costs. Specifically, the activation cost for sensing decisions is defined as a function $C_a : \Theta \to \mathbb{N}$ that assigns a non-negative integer to each sensing decision. The switching cost for each event is a function $c_s : \Sigma \times \{0, 1\} \to \mathbb{N}$, where $(\sigma, 0)$ and $(\sigma, 1)$ denote the cost of turning off the sensor for σ and the cost of turning on the sensor for σ, respectively. Then the switching cost for two sensing decisions is function $C_s : \Theta \times \Theta \to \mathbb{N}$ defined as follows: for any $\theta_1, \theta_2 \in \Theta$, we have

$$C_s(\theta_1, \theta_2) = \sum_{\sigma \in \theta_1 \setminus \theta_2} c_s(\sigma, 0) + \sum_{\sigma \in \theta_2 \setminus \theta_1} c_s(\sigma, 1) \tag{8}$$

That is, $C_s(\theta_1, \theta_2)$ is the switching cost encountered if the sensing decision is switched from θ_1 to θ_2. We assume that the values of the activation cost and the value of the switching cost are both bounded by a non-negative integer C_{max} over their respective domains.

Let ω be a sensor activation policy, s be a string in $\mathcal{L}(G)$, and $\mathcal{R}_\omega(s) := \theta_0\sigma_1\theta_1 \ldots \theta_{n-1}\sigma_n\theta_n$ be its corresponding run as defined in Eq. (3). The cost of string s under ω is defined as the summation the first $|P_\omega(s)|$ activation costs and the first $|P_\omega(s)|$ switching costs, i.e.,

$$C(s, \omega) = \sum_{i=0}^{n-1} C_a(\theta_i) + \sum_{i=0}^{n-1} C_s(\theta_{i-1}, \theta_i) \tag{9}$$

where $\theta_{-1} \in 2^\Sigma$ denotes the initial configuration of the sensors. For the sake of simplicity, we assume that $\theta_{-1} = \emptyset$, i.e., all sensors are off initially (before the system starts). Therefore, the term $C_s(\theta_{-1}, \theta_0) = C_s(\emptyset, \omega(\epsilon))$ represents the switching cost of turning on the system to sensing decision θ_0.

Finally, the cost of a sensor activation policy ω is defined as the *worst-case mean cost* of an infinite run, i.e.,

$$C(\omega) = \limsup_{n \to \infty} \max_{s \in \mathcal{L}(G) : |P_\omega(s)| = n} \{\frac{1}{n} C(s, \omega)\} \tag{10}$$

We now define the optimal sensor activation problem under a quantitative performance objective.

Problem 2. Let G be the system and φ be an IS-based property w.r.t. G. Let C_s and C_a be the switching cost and the activation cost, respectively. Find a sensor activation policy $\omega^* \in \Omega$ s.t. (i) $\omega^* \models_G \varphi$; and (ii) $\forall \omega \in \Omega : \omega \models_G \varphi \Rightarrow C(\omega^*) \leq C(\omega)$.

Although the MPO embeds all sensor activation policies satisfying φ, it cannot be used directly for the purpose of synthesizing a numerically optimal solution, since a Y-state only carries the information of the set of potential states that are possible immediately after the last observation. To address the issue of the switching cost, we must treat Y-states reached from different Z-states, whose sensing decisions are distinct, differently. To this end, we define the Y-augmented BDO as follows.

Given a BDO $\mathcal{O} = (Q^{\mathcal{O}}_Y, Q^{\mathcal{O}}_Z, h^{\mathcal{O}}_{YZ}, h^{\mathcal{O}}_{ZY}, \Sigma, \Theta, y_0)$, its Y-Augmented BDO $\tilde{\mathcal{O}} = (Q^{\tilde{\mathcal{O}}}_Y, Q^{\tilde{\mathcal{O}}}_Z, h^{\tilde{\mathcal{O}}}_{YZ}, h^{\tilde{\mathcal{O}}}_{ZY}, \Sigma, \Theta, \tilde{y}_0)$ is defined by augmenting each Y-state with the sensing decision of its predecessor Z-state. More specifically, we have $Q^{\tilde{\mathcal{O}}}_Z = Q^{\mathcal{O}}_Z$ and $Q^{\tilde{\mathcal{O}}}_Y \subseteq Q^{\mathcal{O}}_Y \times (\Theta \cup \{\emptyset\})$, where $y = (I(y), \Theta(y)) \in Q^{\tilde{\mathcal{O}}}_Y$ if (i) $I(y) \in Q^{\mathcal{O}}_Y$; and (ii) for some $z \in Q^{\mathcal{O}}_Z, \sigma \in \Theta(z)$, we have $h^{\mathcal{O}}_{ZY}(z, \sigma) = I(y)$ and $\Theta(z) = \Theta(y)$. Note that the initial state is $\tilde{y}_0 = (y_0, \emptyset)$ since we assume that all sensors are off initially. Then the transition functions $h^{\tilde{\mathcal{O}}}_{YZ} : Q^{\tilde{\mathcal{O}}}_Y \times \Theta \to Q^{\tilde{\mathcal{O}}}_Z$ and $h^{\tilde{\mathcal{O}}}_{ZY} : Q^{\tilde{\mathcal{O}}}_Z \times \Sigma \to Q^{\tilde{\mathcal{O}}}_Y$ are defined, respectively, as follows: (i) for any $y = (I(y), \Theta(y)) \in Q^{\tilde{\mathcal{O}}}_Y, \theta \in \Theta$, we have $h^{\tilde{\mathcal{O}}}_{YZ}(y, \theta) = h^{\mathcal{O}}_{YZ}(I(y), \theta)$; and (ii) for any $z = (I(z), \Theta(z)) \in Q^{\tilde{\mathcal{O}}}_Z$, $\sigma \in \Theta(z)$, we have $h^{\tilde{\mathcal{O}}}_{ZY}(z, \sigma) = (h^{\mathcal{O}}_{ZY}(z, \theta), \Theta(z))$. Fig. 4 shows the Y-augmented BDO

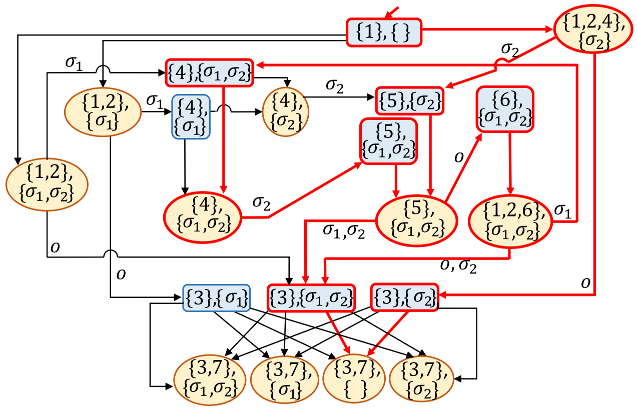


Fig. 4. The Y-augmented BDO of the MPO in Fig. 3. The sensing decision for each transition in $h_{YZ}^{\tilde{\mathcal{O}}}$ is omitted; it can be uniquely determined by the Z-state reached.

of the MPO shown in Fig. 3. Clearly, we see that Y-states $\{3\}$, $\{4\}$ and $\{5\}$ in Fig. 3 are split into several different Y-states in Fig. 4.

Hereafter, we show that Problem 2 can be formulated as a *mean payoff game* on a weighted bipartite graph constructed from the Y-augmented MPO. First, we recall some notions from graph theory and graph games. A weighted bipartite graph is a triple $\mathcal{G} = (V_1\dot{\cup}V_2, E_1\dot{\cup}E_2, \kappa)$, where $V = V_1\dot{\cup}V_2$ is the set of vertices with a unique initial vertex $v_0 \in V_1$, $E = E_1\dot{\cup}E_2 \subseteq (V_1 \times V_2) \cup (V_2 \times V_1)$ is the set of edges and $\kappa : E \to \{-K, \ldots, -1, 0, 1, \ldots, K\}, K \in \mathbb{N}$ is the weight function. A two-player game on $\mathcal{G}$ proceeds as follows. Player 1 first chooses an edge $e_1 \in E_1$ from $v_0 \in V_1$. Then Player 2 chooses an edge $e_2 \in E_2$ from the vertex reached, and so on, indefinitely. The goal of Player 1 is to minimize $\limsup_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\kappa(e_i)$. It was shown in Zwick and Paterson (1996) that there exists a rational number v^*, called the *value of the game*, such that Player 1 has an optimal strategy SG^* such that $\limsup_{n\to\infty}\max_{e\in\mathcal{E}(SG^*):|e|=n}\frac{1}{n}\sum_{i=1}^{n}\kappa(e_i) = v^*$, where $\mathcal{E}(SG^*)$ denotes the set of all possible sequences of edges under strategy SG^*. Moreover, SG^* is *positional*, i.e., its decision only depend on the current vertex reached and it can be represented as a sub-graph of $\mathcal{G}$.

In our problem, Player 1 is the policy and Player 2 is the system. Let $\widetilde{\mathcal{MPO}}_\varphi = (\tilde{Q}_Y, \tilde{Q}_Z, \tilde{h}_{YZ}, \tilde{h}_{ZY}, \Sigma, \Theta, \tilde{y}_0)$ be the Y-augmented MPO. We define a weighted graph $\tilde{\mathcal{G}} = (V_1 \cup V_2, E_1 \cup E_2, \kappa)$ based on $\widetilde{\mathcal{MPO}}_\varphi$ by treating each state in $\tilde{Q}_Y$ (respectively, $\tilde{Q}_Z$) as a vertex in V_1 (respectively, V_2). Then $\langle y, z\rangle \in E_1$ if $\exists\theta \in \Theta : \tilde{h}_{YZ}(y, \theta) = z$ and $\langle z, y\rangle \in E_2$ if $\exists\sigma \in \Sigma : \tilde{h}_{ZY}(z, \sigma) = y$. The weight function κ is defined by:

- For any $e_1 = \langle y, z\rangle \in E_1$, we have $\kappa(e_1) = C_s(\Theta(y), \Theta(z))$.
- For any $e_2 = \langle z, y\rangle \in E_2$, we have $\kappa(e_2) = C_a(\Theta(z))$.

Moreover, for each $z \in V_2$ from which no edge is defined, which corresponds to a Z-state from which no observable event can occur, we add an auxiliary vertex $Y_z \in V_1$ and auxiliary edges $\langle z, Y_z\rangle \in E_2$ and $\langle Y_z, z\rangle \in E_1$, connecting these states, with zero weights for both of these two edges. This auxiliary vertex guarantees that the game graph constructed is complete. Since the values of these auxiliary edges are zero, they will not affect the worst-case mean cost in the limit.

Since $\tilde{\mathcal{G}}$ is constructed from $\widetilde{\mathcal{MPO}}_\varphi$ except the auxiliary vertices, its sub-graph $\mathcal{G}^*$ representing the optimal strategy also corresponds to a deterministic Y-augmented BDO, denoted by $\mathcal{O}^*$, which is a sub-system of $\widetilde{\mathcal{MPO}}_\varphi$. We denote by $\omega_{\mathcal{O}^*}$ the unique sensor activation policy allowed by $\mathcal{O}^*$. The following theorem shows that $\omega_{\mathcal{O}^*}$ actually solves Problem 2.

Theorem 3. *Let SG^* be the optimal positional strategy for the game on $\tilde{\mathcal{G}}$ and v^* be the value of the game. Let $\omega_{\mathcal{O}^*}$ be the sensor activation policy induced by SG^*. Then for any $\omega \in \mathcal{MPO}_\varphi$, we have $C(\omega_{\mathcal{O}^*}) \leq C(\omega)$. Moreover, $C(\omega_{\mathcal{O}^*}) = 2v^*$.*

Proof. Let ω be a sensor activation policy. Then ω defines a strategy SG_ω for Player 1 as follows. Initially, at y_0, it chooses $\langle y_0, z_0\rangle$ such that $y_0 \xrightarrow{\omega(\epsilon)} z_0$. If Player 2 chooses $\langle z_0, y_1\rangle$ at z_0, then this implies that there exists $\sigma_1 \in \Theta(z_0)$ such that $z_0 \xrightarrow{\sigma_1} y_1$. Then Player 1 chooses $\omega(s)$ such that $P_\omega(s) = \sigma_1$ and so forth. Note that strategy SG_ω need not be positional. Then, let $s \in \mathcal{L}(G)$ be a string such that $P_\omega(s) = \sigma_1 \ldots \sigma_n$, and let $y_0 \xrightarrow{\theta_0} z_0 \xrightarrow{\sigma_1} \ldots \xrightarrow{\theta_{n-1}} z_{n-1} \xrightarrow{\sigma_n} y_n$ be the Y- and Z-states reached along s under ω in $\widetilde{\mathcal{MPO}}_\varphi$. The above run also defines a sequence of edges in $\tilde{\mathcal{G}}$ $e_1e_2\ldots e_{2n-1}e_{2n} := \langle y_0, z_0\rangle\langle z_0, y_1\rangle \ldots \langle y_{n-1}, z_{n-1}\rangle\langle z_{n-1}, y_n\rangle$. By the definition of κ, we know that $C(s, \omega) = \sum_{i=1}^{2n}\kappa(e_i)$. Therefore,

$$C(\omega) = \limsup_{n\to\infty}\max_{s\in\mathcal{L}(G):|P_\omega(s)|=n}\{\frac{1}{n}C(s, \omega)\} \tag{11}$$

$$= 2\limsup_{n\to\infty}\max_{e\in\mathcal{E}(SG_\omega):|e|=2n}\{\frac{1}{2n}\sum_{i=1}^{2n}\kappa(e_i)\}$$

Now, let us assume that there exists a sensor activation policy $\omega' \in \mathcal{MPO}_\varphi$ such that $C(\omega') < C(\omega_{\mathcal{O}^*})$. Then we know that

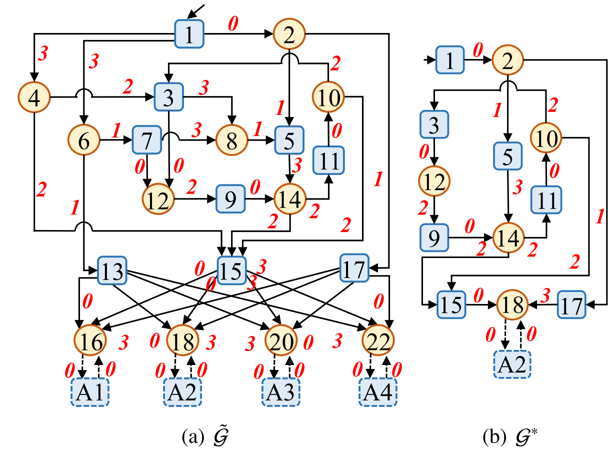
$$\limsup_{n\to\infty}\max_{e\in\mathcal{E}(SG_{\omega'}):|e|=n}\{\frac{1}{n}\sum_{i=1}^{n}\kappa(e_i)\}$$

$$=\frac{C(\omega')}{2} < \frac{C(\omega_{\mathcal{O}^*})}{2} = \limsup_{n\to\infty}\max_{e\in\mathcal{E}(SG^*):|e|=n}\{\frac{1}{n}\sum_{i=1}^{n}\kappa(e_i)\}$$

However, it contradicts the fact that SG^* is the optimal strategy for Player 1 on $\tilde{\mathcal{G}}$. Therefore, no such ω' exists. Finally, by Eq. (11), we also know that $C(\omega_{\mathcal{O}^*}) = 2\limsup_{n\to\infty}\max_{e\in\mathcal{E}(SG^*):|e|=n}\{\frac{1}{n}\sum_{i=1}^{n}\kappa(e_i)\} = 2v^*$. □

Example 6. We return to system G in Fig. 1 and IS-based property φ defined in Example 1. The corresponding MPO is shown in Fig. 3 and the Y-augmented MPO is shown in Fig. 4. For each sensing decision, its activation cost is defined by the number of events in Σ_s we decide to monitor, i.e., $\forall\theta \in \Theta : C_a(\theta) = |\theta \cap \Sigma_s|$. For example, we have $C_a(\{o, \sigma_1\}) = 1$ and $C_a(\{o, \sigma_1, \sigma_2\}) = 2$. We also define the switching costs as $c_s(o, 0) = c_s(o, 1) = 0$, $c_s(\sigma_1, 0) = c_s(\sigma_1, 1) = 3$, and $c_s(\sigma_2, 0) = c_s(\sigma_2, 1) = 0$, i.e., σ_2 can be freely switched without any cost. Note that cost $c_s(o, 1)$ only occurs when the system starts, since o is always observable thereafter.

The graph $\tilde{\mathcal{G}}$ constructed based on the Y-augmented MPO is shown in Fig. 5(a), where each vertex has been renamed for simplicity and the integer associated with each edge denotes its weight. States A1–A4 are four auxiliary edges, since no observable event can occur from vertices 16, 18, 20 and 22. By applying the mean-payoff game algorithm, we find that the value of the game is $v^* = 1$ and the optimal strategy that achieves this value is represented by graph $\mathcal{G}^*$ shown in Fig. 5(b). By removing auxiliary edges from $\mathcal{G}^*$, we obtain the Y-augmented BDO $\mathcal{O}^*$ that realizes the optimal sensor activation policy $\omega_{\mathcal{O}^*}$, which is the highlighted part in Fig. 4. Based on the previous discussion, we know that $C(\omega_{\mathcal{O}^*}) = 2$. Interestingly, $\omega_{\mathcal{O}^*}$ is not a logically minimal solution. For example at Y-state $(\{4\}, \{o, \sigma_1, \sigma_2\})$, we need to take sensing decision $\{o, \sigma_2\}$ in order to achieve logical minimality. However, if we take $\{o, \sigma_2\}$, then the increment of the switching cost is larger than the decrement of the activation cost. This implies that there exists a tradeoff between the activation costs and the switching costs.



(a) $\tilde{\mathcal{G}}$ (b) $\mathcal{G}^*$

Fig. 5. Figures in Example 6.

Remark 5. The Y-augmented BDO of the MPO has at most $2^{|Q|+|\Sigma_s|}$ Y-states and the same number of Z-state. Therefore, the corresponding graph $\mathcal{G}$ has at most $|V|:=2^{|Q|+|\Sigma_s|+1}$ vertices and $|E|:=(2^{|\Sigma_s|}+|\Sigma|)2^{|Q|+|\Sigma_s|}$ edges. The optimal strategy can be found using the algorithm in Brim, Chaloupka, Doyen, Gentilini, and Raskin (2011) with a complexity $O(K|E\parallel V|^2(\log|V|+\log K))$, where $K=C_{max}$.

6. Applications of the generalized MPO

In this section, we show how the generalized MPO can be applied to specific sensor activation problems.

6.1. Application to control and diagnosis

Observability and diagnosability are two key properties of interest in control and diagnosis of DES. It is shown in Wang et al. (2007) that the problem of sensor activation for observability can be formulated as a state-disambiguation problem. Similarly, it is shown in Dallal and Lafortune (2014) that the problem of sensor activation for K-diagnosability can be formulated as a state-disambiguation problem. Therefore, as was discussed in Remark 1, both of these sensor activation problems can be solved by the generalized MPO approach that we have presented. Another property of interest in sensor activation is detectability (Shu, Lin, & Ying, 2007); it is related to state reconstruction. By using the same approach that is used for the reformulation of K-diagnosability in Dallal and Lafortune (2014), we can show that strong K-detectability can also be formulated as an IS-based property and thereby our solution procedure also applies to that property.

6.2. Application to fault prediction

As a specific example of how the methodology presented in this paper can be used to solve problems that have not yet been addressed in the literature, we consider the problem of sensor activation for the enforcement of *predictability*, a notion introduced in Genc and Lafortune (2009). Let $f\in\Sigma$ be the fault event to be predicted. We denote by $\Psi(f):=\{sf\in\mathcal{L}(G):s\in\Sigma^*\}$ the set of strings that end with f. We write $f\in s$ if $\bar{s}\cap\Psi(f)\neq\emptyset$.

Definition 7 (*Predictability*)**.** A live language $\mathcal{L}(G)$ is said to be predictable w.r.t. $f\in\Sigma$ and ω if $(\forall s\in\Psi(f))(\exists t\in\overline{\{s\}}:f\notin t)$ $(\forall u\in\mathcal{L}(G):f\notin u\wedge P_\omega(u)=P_\omega(t))(\exists n\in\mathbb{N})(\forall v\in\mathcal{L}(G)/u)[|v|\geq n\Rightarrow f\in v]$.

The above definition requires that the fault event f should be predicted unambiguously before its occurrence. To proceed further, we assume, w.l.o.g., that state space of G is partitioned into two disjoint sets $Q=Q_Y\dot{\cup}Q_N$, such that (i) $\forall s\in\mathcal{L}(G):\delta(q_0,s)\in Q_Y\Rightarrow f\in s$; and (ii) $\forall s\in\mathcal{L}(G):\delta(q_0,s)\in Q_N\Rightarrow f\notin s$. That is, Q_Y is the set of faulty states and Q_N is the set of non-faulty states. Next, similarly to the notions of boundary strings and indicator strings in Kumar and Takai (2010), we define the set of *boundary states* as $\partial_Q=\{q\in Q:\delta(q,f)!\}$ and the set of *non-indicator states* as $\mathcal{N}_Q=\{q\in Q_N:\forall n\in\mathbb{N},\exists s\in\mathcal{L}(G,q)\text{ s.t. }|s|>n\wedge f\notin s\}$.

With the above notions, we define the IS-based property $\varphi_{pre}:2^Q\to\{0,1\}$ by:

$$\forall i\in 2^Q:[\varphi_{pre}(i)=0]\Leftrightarrow[\exists q,q'\in i:q\in\partial_Q\wedge q'\in\mathcal{N}_Q] \tag{12}$$

The following result says that predictability is equivalent to the IS-based property φ_{pre}. Its proof is available in Yin and Lafortune (0000).

Theorem 4. *Let φ_{pre} be the IS-based property defined by Eq.* (12)*. For any sensor activation policy $\omega\in\Omega$, $\mathcal{L}(G)$ is predictable w.r.t. f and ω if and only if $\omega\models_G\varphi_{pre}$.*

Therefore, to synthesize a minimal sensor activation policy for predictability, it suffices to solve Problem 1 w.r.t. φ_{pre}.

Example 7. Let us return to the system G in Fig. 1. Suppose that f is a fault event. System G already satisfies the state partition assumption $Q=Q_Y\dot{\cup}Q_N$, where $Q_N=\{1,2,3,4,5,6\}$ and $Q_Y=\{7\}$. Also, we have $\partial_Q=\{3\}$ and $\mathcal{N}_Q=\{1,4,5,6\}$. In fact, we see that the IS-based property defined in Example 1 that we considered in the previous examples is indeed the IS-based property φ_{pre} for this example. Therefore, the solutions $\mathcal{O}_1$ and $\mathcal{O}_2$ shown in Fig. 2 that we obtained previously are two (incomparable) logical minimal sensor activation policies that guarantee predictability and $\mathcal{O}^*$, which is the highlighted part of Fig. 4, is the quantitative optimal solution.

6.3. Application to cyber-security

As was discussed earlier in Remark 2, in some cases, the system may also be monitored by an *external* observer that is potentially malicious. We recall an important security property called current-state opacity.

Definition 8. Secret $Q_S\subseteq Q$ is current-state opaque w.r.t. G and ω if $\forall s\in\mathcal{L}(G):\mathcal{E}_\omega^G(s)\not\subseteq Q_S$.

Current-state opacity is clearly an IS-based property. The problem of synthesizing a maximal/optimal sensor activation policy (or dynamic mask) can also be solved by the approach presented in this paper. Moreover, the same approach can be applied to other user defined properties. For example, consider the IS-based property $\varphi_{Kano}:2^Q\to\{0,1\}$ defined by

$$\forall i\in 2^Q:\varphi_{Kano}(i)=0\Leftrightarrow|i|\leq K \tag{13}$$

where $K\in\mathbb{N}$. This property is related to *K-anonymity* studied in the computer security literature (Sweeney, 2002). Intuitively, it requires that the observer should never determine the current-state of the system "too precisely" in the sense that the cardinality of $\mathcal{E}_\omega^G(s)$ is smaller than or equal to K. We can also synthesize a sensor activation policy that guarantees this property.

6.4. Composition of IS-based properties

Finally, we would like to remark that IS-based properties are compositional in the sense that for any two IS-based properties φ_1 and φ_2, $\varphi_1 \wedge \varphi_2 : 2^Q \rightarrow \{0, 1\}$ defined by $(\varphi_1 \wedge \varphi_2)(i) = 1 \Leftrightarrow \varphi_1(i) = 1 \wedge \varphi_2(i) = 1$ is also an IS-based property that captures the satisfactions of both φ_1 and φ_2. Therefore, our framework also allows the enforcement of multiple properties at the same time. For example, if we want to synthesize a sensor activation policy satisfying some utility requirement, e.g., to predict fault occurrences, and at the same time, we also want that the information released by the sensor activation policy should meet certain security or privacy concern, e.g., to keep the system K-anonymous, then it suffices to enforce the IS-based property $\varphi_{pre} \wedge \varphi_{Kano}$ within our framework.

7. Conclusion

We presented a new approach for the problem of synthesizing an optimal sensor activation policy that guarantees some observation property in problems of control, diagnosis, prediction, or other types, in the context of partially-observed discrete event systems. We defined the generalized Most Permissive Observer that is applicable to a wide class of properties called information-state-based properties. Both the problem of synthesizing a logically optimal sensor activation policy and the problem of synthesizing an optimal sensor activation policy with respect to a quantitative cost function were solved based on the generalized MPO. Our approach generalizes the previous works on the MPO, which pertain to specific properties such as opacity or K-diagnosability. Moreover, our approach is applicable to a wide class of user-defined properties.

References

Brim, L., Chaloupka, J., Doyen, L., Gentilini, R., & Raskin, J. -F. (2011). Faster algorithms for mean-payoff games. *Formal Methods in System Design*, *38*(2), 97–118.

Cassandras, C., & Lafortune, S. (2008). *Introduction to discrete event systems* (2nd ed.). Springer.

Cassez, F. (2010). Dynamic observers for fault diagnosis of timed systems. In *49th IEEE conf. decision and control* (pp. 4359–4364).

Cassez, F., Dubreil, J., & Marchand, H. (2012). Synthesis of opaque systems with static and dynamic masks. *Formal Methods in System Design*, *40*(1), 88–115.

Cassez, F., & Tripakis, S. (2008). Fault diagnosis with static and dynamic observers. *Fundamenta Informaticae*, *88*(4), 497–540.

Dallal, E., & Lafortune, S. (2014). On most permissive observers in dynamic sensor activation problems. *IEEE TAC*, *59*(4), 966–981.

Genc, S., & Lafortune, S. (2009). Predictability of event occurrences in partially observed discrete-event systems. *Automatica*, *45*(2), 301–311.

Kumar, R., & Takai, S. (2010). Decentralized prognosis of failures in discrete event systems. *IEEE Transactions on Automatic Control*, *55*(1), 48–59.

Sears, D., & Rudie, K. (2016). Minimal sensor activation and minimal communication in discrete-event systems. *Discrete Event Dynamic Systems: Theory & Applications*, *26*(2), 295–349.

Shu, S., Huang, Z., & Lin, F. (2013). Online sensor activation for detectability of discrete event systems. *IEEE Transactions on Automatic Science and Engineering*, *10*(2), 457–461.

Shu, S., Lin, F., & Ying, H. (2007). Detectability of discrete event systems. *IEEE Transactions on Automatic Control*, *52*(12), 2356–2359.

Sweeney, L. (2002). K-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *10*(05), 557–570.

Thorsley, D., & Teneketzis, D. (2007). Active acquisition of information for diagnosis and supervisory control of discrete event systems. *Discrete Event Dynamic Systems: Theory & Applications*, *17*(4), 531–583.

Wang, W., Lafortune, S., Girard, A., & Lin, F. (2010). Optimal sensor activation for diagnosing discrete event systems. *Automatica*, *46*(7), 1165–1175.

Wang, W., Lafortune, S., & Lin, F. (2007). An algorithm for calculating indistinguishable states and clusters in finite-state automata with partially observable transitions. *Systems & Control Letters*, *56*(9), 656–661.

Wang, W., Lafortune, S., Lin, F., & Girard, A. (2010). Minimization of dynamic sensor activation in discrete event systems for the purpose of control. *IEEE Transactions on Automatic Control*, *55*(11), 2447–2461.

Wu, Y. -C., & Lafortune, S. (2014). Synthesis of insertion functions for enforcement of opacity security properties. *Automatica*, *50*(5), 1336–1348.

Yin, X., & Lafortune, S. (0000). Supplementary material for 'A general approach for optimizing dynamic sensor activation for discrete event systems'. http://xiangyin.sjtu.edu.cn/Paper/general-supp.pdf.

Yin, X., & Lafortune, S. (2015). A general approach for solving dynamic sensor activation problems for a class of properties. In *54th IEEE conference on decision and control* (pp. 3610–3615).

Yin, X., & Lafortune, S. (2016). A uniform approach for synthesizing property-enforcing supervisors for partially-observed discrete-event systems. *IEEE Transactions on Automatic Control*, *61*(8), 2140–2154.

Yin, X., & Lafortune, S. (2018). Minimization of sensor activation in decentralized discrete-event systems. *IEEE Transactions on Automatic Control*, *63*(11), 3705–3718.

Zhang, B., Shu, S., & Lin, F. (2015). Maximum information release while ensuring opacity in discrete event systems. *IEEE Transactions on Automation Science and Engineering*, *12*(4), 1067–1079.

Zwick, U., & Paterson, M. (1996). The complexity of mean payoff games on graphs. *Theoretical Computer Science*, *158*(1), 343–359.

Xiang Yin was born in Anhui, China, in 1991. He received the B.Eng. degree from Zhejiang University in 2012, the M.S. degree from the University of Michigan, Ann Arbor, in 2013, and the Ph.D. degree from the University of Michigan, Ann Arbor, in 2017, all in electrical engineering.

Since 2017, he has been with the Department of Automation, Shanghai Jiao Tong University, where he is an Associate Professor. His research interests include formal methods, control of discrete-event systems, model-based fault diagnosis, security and their applications to cyber and cyber–physical systems. Dr. Yin received the Outstanding Reviewer Awards from Automatica, the IEEE Transactions on Automatic Control and the Journal of Discrete Event Dynamic Systems. Dr. Yin also received the IEEE Conference on Decision and Control (CDC) Best Student Paper Award Finalist in 2016. He is the co-chair of the IEEE CSS Technical Committee on Discrete Event Systems.

Stéphane Lafortune received the B.Eng. degree from Ecole Polytechnique de Montréal in 1980, the M.Eng. degree from McGill University in 1982, and the Ph.D. degree from the University of California at Berkeley in 1986, all in electrical engineering. Since September 1986, he has been with the University of Michigan, Ann Arbor, where he is a Professor of Electrical Engineering and Computer Science.

Dr. Lafortune is a Fellow of the IEEE (1999). He received the Presidential Young Investigator Award from the National Science Foundation in 1990 and the George S. Axelby Outstanding Paper Award from the Control Systems Society of the IEEE in 1994 (for a paper co-authored with S.-L. Chung and F. Lin) and in 2001 (for a paper co-authored with G. Barrett). Dr. Lafortune's research interests are in discrete event systems and include multiple problem domains: modeling, diagnosis, control, optimization, and applications to computer and software systems.

He is the lead developer of the software package UMDES and co-developer of DESUMA with L. Ricker.

He co-authored, with C. Cassandras, the textbook Introduction to Discrete Event Systems — Second Edition (Springer, 2008). Dr. Lafortune is Editor-in-Chief of the Journal of Discrete Event Dynamic Systems: Theory and Applications.