2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Selecting Observation Platforms for Optimized Anomaly Detectability under Unreliable Partial Observations

Wen-Chiao Lin, Humberto E. Garcia, and Tae-Sic Yoo

Abstract— **Diagnosers for keeping track on the occurrences of special events in the framework of unreliable partially-observed discrete-event dynamical systems were developed in previous work. This paper considers observation platforms consisting of sensors that provide partial and unreliable observations and of diagnosers that analyze them. Diagnosers in observation platforms typically perform better as sensors providing the observations become more costly or increase in number. This paper proposes a methodology for finding an observation platform that achieves an optimal balance between cost and performance, while satisfying given observability requirements and constraints. Since this problem is generally computational hard in the framework considered, an observation platform optimization algorithm is utilized that uses two greedy heuristics, one myopic and another based on projected performances. These heuristics are sequentially executed in order to find best observation platforms. The developed algorithm is then applied to an observation platform optimization problem for a multi-unit-operation system. Results show that improved observation platforms can be found that may significantly reduce the observation platform cost but still yield acceptable performance for correctly inferring the occurrences of special events.**

I. Introduction

This paper considers the monitoring architecture shown in Fig. 1 for event occurrence analysis of discrete-event dynamical systems (DEDS). An observation platform here

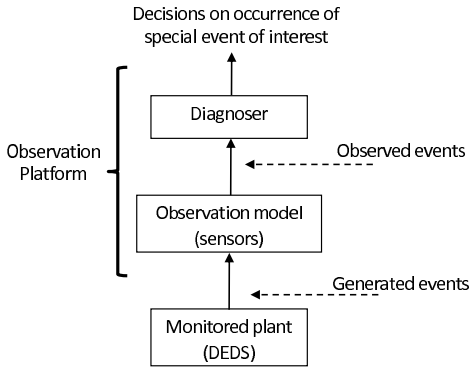

Fig. 1. Monitoring architecture

consists of sensors generating the observed events and a diagnoser that analyzes the data. In previous work, various algorithms for implementing the diagnoser in Fig. 1 have been developed. In particular, the work in [1] deals with detection of special events assuming that a finite-state automaton describes the DEDS, that sensors are reliable, and that failures/faults are permanent. This work is extended in [2] for diagnosing behaviors of interest in discrete event systems. Subsequent extensions and improvements of [1] address the problem of the detection of special events accounting for sensor unreliability and stochastic aspects in DEDS [3]–[9]. Counting of the occurrences of intermittent or non-persistent faults is addressed in [10]–[17]. In particular, the issue of detecting whether or not a resetting has occurred is addressed in [10], and [11] addresses fault counting problems and introduces several notions of diagnosability that capture the various counting capabilities of special events. Event counting assuming a deterministic finite-state automaton with partial observations is addressed in [12]. While [13], [14] present a deterministic counting strategy for accommodating stochastic automata with unreliable observations, [15]–[17] develop algorithms that fully utilize the probabilistic aspects of stochastic automata.

The work mentioned above focuses on developing diagnoser algorithms for analyzing the behavior of DEDS (e.g., detecting special event occurrences) observed through sensor configurations. The costs of sensor configurations may vary with the number of sensors deployed, their quality, their impact on operation, and difficulty of installation, for example. In particular, sensor configurations that cost more may include more sensors and of sensors with better quality, and, typically, they give rise to better diagnoser performance. This paper considers observation platforms, which consists of sensors and of diagnosers. The goal is to propose a simulation-based design methodology for finding optimal observation platforms that balance the cost of the sensors and the performance of a given diagnoser, while satisfying cost constraints and performance requirements. Without loss of generality, the diagnosers considered in this paper are implemented using sequential window diagnosers (SWDs) [16], where the monitored DEDS is modeled as a stochastic automaton under unreliable partial observations. The optimization problem considered here falls into the category considered in [18], namely, that it is computationally hard, and a observation platform optimization algorithm based on heuristics is consequently used for solving it. The algorithm uses two greedy heuristics, one myopic and another that contrarily considers projected sensor performances. The latter heuristic is similar to the approach for optimal sensor selection developed in [19] for heterogeneous sensor networks. However, it has been accordingly modified to address the problem considered here. This algorithm is further extended here from that presented in [20] in order to address an

W.-C. Lin, H.E. Garcia, and T. Yoo are with the Idaho National Laboratory, P.O. Box 1625, Idaho Falls, ID 83415-3675, USA. Email: {Wen-Chiao.Lin, Humberto.Garcia, Tae-Sic.Yoo}@inl.gov

observation platform optimization problem on a multi-unit-operation system.

Other work on optimal sensor selection for DEDS includes [18], [21]–[24]. While [18] addresses computational issues regarding sensor selection for DEDS modeled as finite automata, [21], [22] consider optimal sensor selection for satisfying observability properties in Petri nets. Reference [23] considers the optimization problem in the framework of Fig. 1 assuming, unlike this paper, that sensors are reliable. Optimal sensor selections for supervisory control are addressed in [24].

The main contribution of this paper is extending the results of [20] as follows:

- develop a general methodology for optimizing observation platforms;
- application of optimization algorithms reported in [20] to SWDs described in [16].

The rest of this paper is organized as follows. Section II gives a brief review of the monitoring architecture shown in Fig. 1. The obsevation platform optimization problem is formulated in Section III. A review of the SWD algorithm is also included in Section III, while an observation platform optimization algorithm is developed in Section IV. In Section V, the optimization algorithm is applied to a multi-unit-operation system. Section VI concludes the paper. We assume the reader is familiar with the terminology typical of DEDS.

II. Preliminaries

A. Monitored plant and observation model

The monitored plant in Fig. 1 is modeled as a stochastic automaton, $SA = (X, \Sigma, a, \pi_0)$, where $X := \{x_1, x_2, \ldots, x_{n_x}\}$ is the finite state space, $\Sigma := \{\sigma_1, \sigma_2, \ldots, \sigma_{n_\sigma}\}$ is the set of events, and $\pi_0 := \{\pi_0(x_i) : x_i \in X\}$ is the initial probability distribution of the system. The state transition probability function a is defined as $a : X \times \Sigma \times X \to [0, 1]$, where, $a(x_i, \sigma, x_j)$ denotes the conditional probability that, given the system is in state $x_i \in X$, $\sigma \in \Sigma$ occurs and transitions the system to state $x_j \in X$. Moreover, to insure that the system is live, we assume $\forall x \in X$,

$$\sum_{i=1}^{n_\sigma} \sum_{j=1}^{n_x} a(x, \sigma_i, x_j) = 1, \tag{1}$$

i.e., the occurrence of a new transition is certain from every state. The interest here is to detect and count the number of occurrences of a special event $f \in \Sigma$, which may represent a fault or an anomaly.

As shown in Fig. 1, diagnosers are used to accomplish this task using observations from unreliable sensors. Assume that there is a given pool of available sensors, $U = \{s_1, s_2, \ldots, s_p\}$, from which sensors are chosen for observing the monitored DEDS. Let $\Delta := \{y_1, y_2, \ldots, y_{n_y}\}$ be the set of distinctive observation symbols generated from a sensor configuration $S \subseteq U$. We denote the set of observation symbols at the sensor outputs as

$$\Delta_* := \Delta \cup \{\epsilon\}, \tag{2}$$

where the symbol ϵ indicates that an event has been executed but no observation is reported. The event output function $b : \Sigma \times \Delta_* \to [0, 1]$ satisfies the following: $\forall \sigma \in \Sigma$,

$$b(\sigma, \epsilon) + \sum_{i=1}^{n_y} b(\sigma, y_i) = 1. \tag{3}$$

The functional value $b(\sigma, y)$ is the conditional probability of having output $y \in \Delta_*$ when the system executes event $\sigma \in \Sigma$. For example, assume that the given monitored system generates an event σ. Characterizing $b(\sigma, \epsilon)$, $b(\sigma, \sigma)$, and $b(\sigma, \delta)$ (with $\sigma \neq \delta$) equal to 0.1, 0.7, and 0.2 indicates the probabilities of misdetection, correct classification, and misclassification, respectively, for this sensor. The set of observation symbols, Δ, and the function b both depend on the sensor configuration, S. To further illustrate the observation model, suppose that SA executes the following event sequence: $s = \sigma^1\sigma^2 \ldots \sigma^n \ldots \in \Sigma^*$. Given s and sensor configuration, $S \subseteq U$, there are many possible sequences of output symbols for the unreliable observations modeled by (2) and (3). A particular sequence of output symbols can be denoted by $o = o^1 o^2 \ldots o^n \ldots \in (\Delta_*)^*$, where $b(\sigma^i, o^i) > 0$ for $i > 0$. The sequence of observations available to the diagnoser in Fig. 1 is denoted by $y = y^1 y^2 \ldots y^m \ldots \in \Delta^*$, where $P_\Delta(o) = y$ and $P_\Delta : (\Delta_*)^* \to \Delta^*$ is a plain projection function that removes the silent event ϵ from o. Hence, y is the output sequence o with the symbol ϵ eliminated. Note that y_i denotes the ith symbol in Δ, while y^i denotes the ith observed symbol corresponding to the sequence of generated events.

III. Problem Formulation

Consider the monitoring architecture discussed in Section II and suppose there are N_s special events to detect and count. Let $O = (S, D)$ denote the observation platform, where S is the sensor configuration and D is the diagnoser. The observation platform optimization problem is formulated as follows:

$$O^* := \arg\min\{I(O) : O \subseteq (U, \mathcal{D})\}, \tag{4}$$

subject to $\beta^i(O) \leq \beta^{i*}$, for $i = 1, 2, \ldots, N_s$, and $ct(O) \leq ct^*$, where

- U is the set of all possible sensors considered;
- $\mathcal{D}$ is the set of all diagnosers considered;
- the loss index is defined as

$$I(O) = \sum_{i=1}^{N_s} c_i \cdot \beta^i(O) + c \cdot ct(O); \tag{5}$$

- $\beta^i(O)$, $i = 1, 2, \ldots, N_s$, is the performance measure for detecting and counting the ith special event;
- $ct(O)$ is the cost of the observation platform;
- ct^* is the maximum cost desired for O;

- β^{i*} indicates the desired performance for the diagnoser in estimating the number of occurrences of the ith special event.

Details of the cost and performance measure of the observation platform are addressed in the following subsections.

A. Observation platform cost

For a observation platform, $O = (S, D) \subseteq (U, \mathcal{D})$, the cost is given by:

$$ct(O) = \sum_{s \in S} ct(s) + ct(D), \tag{6}$$

where $ct(s)$ and $ct(D)$ denote the costs of a particular sensor $s \in S$ and for implementing the diagnoser D. The sensor cost, $ct(s)$, is a compound measure for sensor s that includes metrics such as monetary cost, vulnerability, and intrusiveness of operation. On the other hand, the diagnoser cost, $ct(D)$ is typically a single, fixed figure associated with the cost of implementing a diagnoser in a computer software. Without loss of generality but for simplicity, let's assume that $ct(D) = 0$ and (6) reduces to $ct(O) = \sum_{s \in S} ct(s)$.

B. Observation platform performance

The performance measure of the observation platform is described here. To simplify the presentation, assume there is one special event, f, to be counted. Let $\beta(O)$ denote the performance of the observation platform, O. This performance measure is determined by $\beta(O) = F(P_M(O), P_F(O))$, where F is a function depending on the probabilities of misdetection (i.e., the probability that f occurred but the occurrence was not counted, denoted by $P_M(O)$) and false alarm (i.e., the probability that f did not occur but an occurrence was counted, denoted by $P_F(O)$), which in turn depend on O. To elaborate, suppose α is a general design parameter for the diagnoser D in O and consider specific observation windows to evaluate $P_M(O)$ and $P_F(O)$. The parameter α typically presents a trade off between $P_M(O)$ and $P_F(O)$. For example, α may be such that, when it becomes large (or small), the probability of misdetection becomes small (or large) while the probability of false alarm becomes large (or small). As the value of α varies, values of $P_M(O)$ and $P_F(O)$ can be calculated (via experiments or simulations). Fig. 2 shows two typical curves generated by plotting $P_M(O)$ against $P_F(O)$ as α is varied. These are the misdetection-false alarm tradeoff curves corresponding to two different observation platforms. Note that the observation platform corresponding to the solid curve gives a better misdetection-false alarm tradeoff than the one corresponding to the dashed curve. That is, for the same value of $P_F(O)$, the solid curve gives a smaller value of $P_M(O)$ than the dashed curve. The optimal case is a curve that reduces to a point at the origin (i.e., regardless of the selected value for α, its probabilities for false alarm and misdetection are both zero). Hence, in general, a curve closer to the origin indicates better observation platform performance. A measure of how close the curve is to the origin is calculated by the area under the curve. Let $Area(O)$ denote the area under the $P_M(O) - P_F(O)$ curve for observation platform O. Then, $\beta(O) = F(P_M(O), P_F(O)) = Area(O)$. Note that when there are no sensors in the observation platform (hence no observation), it is impossible to count the number of occurrences of f, even under the assumption of having the discrete event model of the monitored system. In this case, we set $\beta(O) = \infty$. We mention that the best value of α for the diagnoser can then be chosen based on the $P_M(O) - P_F(O)$ curve according to user specified observation requirements (i.e., values of $P_F(O)$ and $P_M(O)$). Note also that because our performance measure is a function of broadly-used statistical characterizations (i.e., probabilities of false alarm and misdetection), experimental realizations are needed to estimate their values.

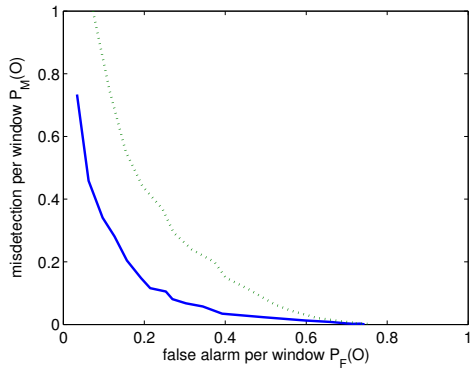

Fig. 2. Misdetection-false count trade-off curve

In this paper, we consider using SWD as the diagnoser for the observation platform, although other types of diagnoser algorithms can readily be accommodated. In addition, while we do focus here on solving (4) by optimally selecting a sensor configuration S for O, we do not consider selecting the best diagnoser algorithm from several possible ones; instead, we focus on identifying an optimal value for its design parameter (i.e., α), while keeping the algorithm for implementing D fixed to that of SWD. In general, when several candidate diagnoser algorithms (in $\mathcal{D}$) are considered, an additional optimization loop is needed to select best overall diagnoser algorithm based on estimated performances. However for simplicity, this step is omitted here. A brief review of SWDs is given below.

Suppose the special event, f, is to be detected and counted. As the observations, y^i, $i = 1, 2, \ldots$, become available from the sensors, the SWD estimates the number of times f has occurred. In particular, when the mth observation becomes available, the count estimate is calculated based on the observations y^1 through y^m and is denoted by $c(m)$. Let $m \geq 0$ and define the state probability vector,

$$\phi(m) = \left[p_1^N(m), p_2^N(m), \ldots, p_{n_x}^N(m), p_1^F(m), \ldots, p_{n_x}^F(m)\right], \tag{7}$$

where $p_i^N(m)$ (or $p_i^F(m)$) denotes the conditional probability that the DEDS is in state $x_i \in X$ and that f has not been (or has been) executed given m available observations and

given initial probability

$$\phi(0) = [\pi_0, 0, 0, \ldots, 0]. \tag{8}$$

In order to update the state probability vectors as observations become available, a set of $2n_x \times 2n_x$ diagnoser matrices, is constructed. This set of matrices is denoted by $\mathcal{X}$ and has a one-to-one correspondence with the observation symbols in Δ. Hence, for each $y \in \Delta$, there is exactly one matrix, $\Phi^{SD}(y) \in \mathcal{X}$. The general structure of $\Phi^{SD}(y)$ is

$$\Phi^{SD}(y) = \begin{bmatrix} \Phi^{NN}(y) & \Phi^{NF}(y) \\ 0_{n_x \times n_x} & \Phi^{FF}(y) \end{bmatrix}, \tag{9}$$

where $0_{n_x \times n_x}$ is the $n_x \times n_x$ zero matrix and $\Phi^{NF}(y)$ (or $\Phi^{NN}(y)$) is an $n_x \times n_x$ matrix, which determines the probability that the system, starting at a state where f has not occurred before, transitions to a state $x_i \in X$ where f has (or has not) occurred given the observation y. Likewise, $\Phi^{FF}(y)$ determines the probability that the system, starting at a state where f has occurred before, transitions to a state $x_i \in X$ where f has occurred given the observation y. While detailed descriptions of these matrices and their computations can be found in [5], [16], the procedure for updating $\phi(m)$ is described below for completeness.

When there is no observations available yet (i.e., $m = 0$), the state probability vector is set as in (8) and the estimated count, $c(0)$, is set to 0. As the $(m+1)$st observation (i.e., y^{m+1}) becomes available, the SWD updates $\phi(m+1)$ according to

$$\phi(m+1) = \frac{\phi(m)\Phi^{SD}(y^{m+1})}{\|\phi(m)\Phi^{SD}(y^{m+1})\|}, \tag{10}$$

where $\| \bullet \|$ denotes the 1-norm. For more details of the update in (10), we refer the readers to [16]. Let $0 < \alpha < 1$ be a user selected parameter, referred to as the false alarm tolerance, and consider the following. If

$$\sum_{i=1}^{n_x} p_i^F(m+1) > 1 - \alpha, \tag{11}$$

i.e., the probability that f has occurred is greater than $1-\alpha$, increase the estimated count of occurrence of f, i.e., $c(m+1) = c(m) + 1$, and *reset* $\phi(m+1)$ such that

$$p_i^N(m+1) = p_i^N(m+1) + p_i^F(m+1), \tag{12}$$

and

$$p_i^F(m+1) = 0, \tag{13}$$

for $1 \le i \le n_x$. If (11) does not hold, set $c(m+1) = c(m)$. This process is repeated as observations become available. Note that the reset defined by (12) and (13) means that, after an increase in the estimated count, the SWD resets the probability of f occurring to 0 while still keeps track of the probabilities for the DEDS to be in state $x_i \in X$, $i = 1, 2, \ldots, n_x$. Through the described procedure above, the SWDs sequentially estimates the number of times the special event f has occurred given the observations.

Note that the user selected parameter α strongly influences the performance of the SWD. If a small α is chosen, (11) is harder to be satisfied and the SWD tends to under count the occurrences of f. Conversely, the SWD tends to over count for large α. Another way of viewing this is to consider a given window of events that occurred in between an increase of estimated count. The number of misdetections per window increases when α becomes small, and, conversely, false counts per window increases when α becomes large. Note that, for given α, the number of misdetections and false counts per window can be calculated via off-line simulations with the monitored DEDS executing a sequence of n events. The number of misdetections and false counts per window can be plotted against each other as α sweeps through the interval $(0, 1)$ with small increments (e.g., $\alpha = 0.05, 0.15, \ldots, 0.95$). Simulations for calculating the misdetections and false counts start anew for each value of α considered. Fig. 2 was actually generated in this way using SWD as the diagnoser.

C. Extensions to multiple special events

The description of the diagnosers above considers only one special event, f. Extension in the sense of [1] to estimate the occurrences of multiple events is straightforward. In particular, let the special events be labeled as f_i, $i = 1, 2, \ldots, N_s$, where N_s is the number of special events. The design parameter chosen for f_i (e.g., false alarm tolerance of the SWD designed for f_i) is denoted by α_i. Given a specific observation platform, O, the performance corresponding to f_i is denoted by $\beta^i(O)$.

IV. Solution to the Observation Platform Optimization Problem

Given the structure of the diagnoser D, the observation platform optimization problem in (4) is solved by adding sensors sequentially to the observation platform. A myopic criterion may be considered for this procedure. However, the myopic criterion does not often perform well. An example is provided in [20] to explain this limitation, with a brief explanation included below for completeness. In short, if the goal is to find the least costly observation platform that satisfies observability constraints, the resulting observation platform may consist of many cheap sensors with typical poor performance. However, there may exist a less costly observation platform that satisfies the observability requirements by relying on few expensive sensors, but with good performances.

In the following, the optimization problem is solved by first using the myopic criterion to pick only the first sensor to be added. Then, subsequent sensors are added using a projected optimal loss index sensor selection criterion. Detailed descriptions of the two sensor selection criteria are provided in the following subsections.

A. Myopic criterion

To pick the first sensor to be added, the current observation platform, $O_{cur} = (S_{cur}, D)$, is set to $(\emptyset, D)$, where D is the

diagnoser considered (i.e., an SWD for this paper). A sensor is then added to O_{cur} based on the following criterion: Find

$$s^* = \arg \min_{s \in (U \setminus S_{cur})} \{\sum_{i=1}^{N_s} c_i \cdot \beta^i(O_{cur} \oplus \{s\}) + c \cdot ct(O_{cur} \oplus \{s\})\}, \quad (14)$$

subject to

$$ct(O_{cur} \oplus \{s\}) \leq ct^*, \quad (15)$$

where the operation $O_{cur} \oplus s$ means $(S_{cur} \cup \{s\}, D)$. The sensor, s^*, is the newly selected sensor to be added to O_{cur}. Furthermore, (15) means that observation platforms costing more than ct^* are not considered. This criterion is based on the instantaneous performance improvement of the candidate sensor and is myopic.

B. Projected optimal loss index sensor selection criterion

To explain the projected optimal loss index sensor selection criterion, consider the projected loss index, $I(O_{cur} \oplus \{s_{1:k}\}) = \sum_{i=1}^{N_s} c_i \cdot \beta^i(O_{cur} \oplus \{s_{1:k}\}) + c \cdot ct(O_{cur} \oplus \{s_{1:k}\})$, where

- $O_{cur} \oplus \{s_{1:k}\}$, $k \geq 1$, represents a *fictitious* observation platform, $O_{cur} \oplus s \oplus s \oplus \ldots \oplus s$, where the sensor, s, is added k times to the current observation platform. This fictitious observation platform represents the situation where k sensors with the same statistical performances and costs as s are added to O_{cur}. The idea is to minimize the chance of getting stuck at a local minimum by projecting the value of the loss index assuming that sensors were added with the same statistical performances and costs as the candidate sensor being considered.
- $ct(O_{cur} \oplus \{s_{1:k}\})$ represents the *projected* cost of the fictitious observation platform: $ct(O_{cur} \oplus \{s_{1:k}\}) = \sum_{s' \in S_{cur}} ct(s') + k \cdot ct(s)$.
- $\beta^i(O_{cur} \oplus \{s_{1:k}\})$ represents the *projected* performance of the fictitious observation platform. There is no known closed form expression for calculating the projected performance, and, hence, the heuristic formula,

$$\beta^i(O_{cur} \oplus \{s_{1:k}\}) = \left(\frac{\beta^i(O_{cur} \oplus \{s\})}{\beta^i(O_{cur})}\right)^k \beta^i(O_{cur}), \quad (16)$$

  is used. This formula indicates that, as more sensor s are used, the projected performance of the observation platform improves exponentially.

The projected optimal loss index sensor selection criterion is then formulated as

$$s^* = \arg \min\{I(O_{cur} \oplus \{s_{1:k^*(s)}\}) : s \in U \setminus S_{cur} \text{ and } k^*(s) \text{ defined}\}, \quad (17)$$

where

- $I(O_{cur} \oplus \{s_{1:k^*(s)}\}) = \sum_{i=1}^{N_s} c_i \cdot \beta^i(O_{cur} \oplus \{s_{1:k^*(s)}\}) + c \cdot ct(O_{cur} \oplus \{s_{1:k^*(s)}\})$ is the *projected* optimal loss index;
- $k^*(s)$ minimizes the projected loss index given O_{cur} and s: $k^*(s) = \arg \min_{k \in [k_l(s), k_u(s)]} \{\sum_{i=1}^{N_s} c_i \cdot \beta^i(O_{cur} \oplus \{s_{1:k}\}) + c \cdot ct(O_{cur} \oplus \{s_{1:k}\})\}$;
- $k_u(s)$ is the maximum number of s that can be added to O_{cur} without $O_{cur} \oplus \{s_{1:k}\}$ exceeding ct^*: $k_u(s) = \arg \max_{k \geq 0} \{ct(O_{cur} \oplus \{s_{1:k}\}) \leq ct^*\}$;
- $k_l(s)$ is the least number of s that should be added for the performance of $O_{cur} \oplus \{s_{1:k}\}$ to meet all specified β_i^*: $k_l(s) = \arg \min_{k \geq 0} \left\{\wedge_{i=1}^{N_s} \beta^i(O_{cur} \oplus \{s_{1:k}\}) \leq \beta^{i*}\right\}$.

The algorithm for finding an optimal observation platform is executed by first using the myopic criterion to select the first sensor (assuming SWD as the algorithm for D using its best identified α). Subsequent sensors are selected according to the optimal loss index sensor selection criterion described above. The reason for this arrangement is that, since $\beta^i(\emptyset) = \infty$, for $i = 1, 2, \ldots, N_s$, we cannot compute the projected performance via (16) in order to select the first sensor. While other selection criteria may be used for selecting the first sensor, the current algorithm works well for most practical cases considered. The implementation of the proposed simulation-based optimization algorithm is similar to the one reported in [20] with notations changed accordingly and, hence, its description is omitted here. Notice that it is a simulation-based methodology as it uses statistical properties (i.e., probabilities of false alarm and misdetection) that are estimated via simulation.

V. Application

A. Multi-unit-operation monitored system

For the proposed methodology to be practical, it needs to be applicable to real-world systems. To this end, the methodology is applied to the multi-unit-operation system shown in Fig. 3, which has been derived from an actual facility application. The unit operations UO_i of the system are labeled by integers 1 through 6. The input ports are squares marked by $I1$, $I2$, and $I3$, while the output ports are marked by $O1$, $O2$, $O3$, and $O4$. The symbols F_i, $i = 1, 2, \ldots, 13$, stand for material flow, which may be a discrete item (e.g., container) or fluid (e.g., solution). The hexagons denote twenty one possible sensors.

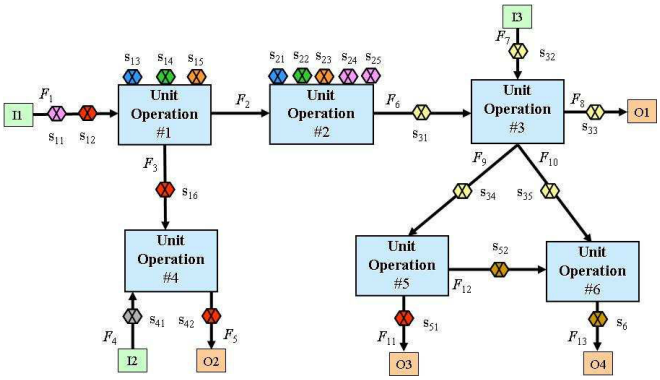

Fig. 3. A multi-unit-operation system

The operations of the system are described below. Input material, F_1, enters the monitored plant via $I1$ and is transferred to UO_1. The outputs of UO_1, F_2 and F_3, are then transferred to UO_2 and UO_4, respectively. Batches of F_3 are stored and processed at UO_4, which are eventually outputted via F_5 through $O2$ after receiving F_4 at UO_4. Likewise, at UO_2, measurements are taken to characterize input material, and the output, F_6, is transferred to UO_3 where it interacts with F_7 entering via $I3$. While UO_3 always outputs F_8 to $O1$, it outputs either F_9 to UO_5 or F_{10} to UO_6. Material in UO_5 can either be transferred out of the monitored plant via $O3$ (i.e., F_{11}) or transferred to UO_6 (i.e., F_{12}). Materials transferred to UO_6 are eventually removed from the monitored plant via $O4$ (i.e., F_{13}).

B. Possible sensors for monitored system

A number of sensors may be deployed to monitor the multi-unit-operation system described above. They are categorized in Table I along with their possible observations.

TABLE I

POSSIBLE SENSORS FOR CONSIDERED MONITORED SYSTEM

Sensors	Discrete event observations (Σ)		
s_{13}, s_{21}	low	normal	
s_{15}, s_{23}	low	normal	high
s_{14}, s_{22}	low	normal	high
s_{12}, s_{16} s_{42}, s_{51}	low	normal	high
s_{31}, s_{32} s_{33}, s_{34} s_{35}	low	normal	high
s_{52}, s_6	transfer	no transfer	
s_{41}	transfer	no transfer	
s_{25} (s_{11} & s_{24})	normal	abnormal	

C. Anomaly patterns of operations

While any anomaly pattern can be detected and tracked by the proposed monitoring system, two anomaly patterns of operations, which represent actual operations that may cause undesirable material to exit the system improperly, are considered. They are described here assuming that the sensors are perfect.

Anomaly pattern $A1$:

A *normal* or *high* property value is provided by s_{12} *and* a *low* property indication is provided by s_{15} *and* a *high* property indication is provided by s_{16} *and* then there is an abnormal event generated by s_{25} (which compares measurements taken by s_{11} and s_{24}). If after three (3) or more instances of these anomalies have been observed, a *high* property indication is provided by s_{42} but *no* observation was received from s_{41}, an alarm is triggered.

Anomaly pattern $A2$:

A *low* property indication is provided by s_{31} *and* a *low* or *normal* property indication is provided by s_{33} *and* a *high* property indication is provided by s_{34}. If after three (3) or more instances of these anomalies have been observed, a *high* property indication is provided by s_{51} is received, an alarm is triggered.

D. Modeling of monitored system

In the interest of space, the procedure for modeling the multi-unit-operation system as a DEDS is omitted but outlined here. Each unit operation is first modeled as an automaton. With a slight abuse of notations, let UO_i, $i = 1, 2, \ldots, 6$ indicate the automata corresponding to the unit operations in Fig. 3. The technique introduced in [12] is employed to model the anomaly patterns for detecting and counting. For each anomaly pattern to be detected, an automaton is constructed, where a fictitious unobservable event is executed if the anomaly pattern of operations occur. Let AP_1 and AP_2 denote the automata for anomaly patterns $A1$ and $A2$, respectively, and f_1 and f_2 be their respective fictitious unobservable events. The global system model is constructed by combining the unit operation models, UO_i, $i = 1, 2, \ldots, 6$, AP_1, and AP_2 using the parallel composition described in [25][1]. The state transition probabilities of the global model is chosen suitable for the simulation study below. In practice, the transition probabilities may be obtained based on past observations. The goal of the diagnoser in Fig. 1 is to infer (given partial and unreliable observations) the number of occurrences of f_1 and f_2, which correspond to the occurrences of anomaly patterns $A1$ and $A2$, respectively. Notice that only the operations of UO_i, $i = 1, 2, 4$, are relevant to $A1$. Similarly, only UO_i, $i = 3, 5, 6$, are relevant to $A2$. Hence, the diagnoser for the detecting and keeping track of the occurrences of the two anomaly patterns can be constructed modularly.

E. Sensor reliability and cost assumptions

For the purpose of showing the simulation results later in this section, arbitrary values are selected not only for probabilities of misdetections, correct classifications, and misclassifications of available sensors for selection but also for their costs. These values are indicated in Table II, with the probability of misclassification for each particular sensor being equally distributed among all possible misclassifications. For instance, consider s_{14}. The cost associated with this sensor is 3, which is relatively low when compared to others. Likewise, consider that the "true" value of the property being measured is "high" at a given time instance. Then, the probability that s_{14} does not give a reading (misdetection) is 0.02 and the probabilities for it to read "high" (correct classification) or another value (misclassification) are 0.94 and 0.04 (which is equally split among "normal" and "low"), respectively. From the descriptions of the sensors above, s_{11} and s_{24} are treated as one and their combined cost and characteristic are shown in the last row of the table.

F. Performance measure

For each proposed O, the performance measure, $\beta^i(O)$, was used for finding a solution to the observation platform

[1]Note that the addition of AP_1 and AP_2 in the parallel composition does not change the behavior described by UO_1 through UO_6. The only effect is that f_1 and f_2 are executed if and only if $A1$ and $A2$ occur, respectively.

TABLE II

SENSOR RELIABILITY

Sensors	Cost	Prob. of mis-detection	Prob. of correct classification	Prob. of mis-classification
s_{13}, s_{21}	3	0.03	0.94	0.03
s_{15}, s_{23}	1	0.02	0.94	0.04
s_{14}, s_{22}	3	0.02	0.94	0.04
s_{12}, s_{16} s_{42}, s_{51}	7	0.02	0.94	0.04
s_{31}, s_{32} s_{33}, s_{34} s_{35},	11	0.02	0.94	0.04
s_{52}, s_6	4	0.03	0.97	0
s_{41}	5	0	1	0
s_{11}, s_{24}, and s_{25} combined	11	0.03	0.94	0.03

optimization problem. In order to evaluate the on-line performance of SWDs for given O and α_i, the following statistics is employed instead:

$$ANE^i(O) = \frac{1}{n_i}\sum_{k=1}^{n_i}\frac{|k-\hat{T}_i(k)|}{k}, \qquad (18)$$

where n_i is the number of occurrences of the given special event, f_i, at the end of the simulation. Moreover, $\hat{T}_i(k)$ denotes the estimated number of its occurrence just after the given special event has been executed the kth time. It is important to point out that, as α_i is a selectable design parameter for SWD, the $ANE^i(O)$ reported for a given observation platform is computed by selecting α for the given diagnoser according to user observability specifications.

G. Simulation results

To compare and evaluate observation platforms, the observation platform in Fig. 3, which includes all available sensors, is considered as the baseline. The cost of the baseline observation platform is the highest among all possible observation platforms, while its performance is expected to be the best. Any observation platform with comparable performance to the baseline but with significantly lower cost is considered an improvement. In order to find an improved observation platform, the algorithm proposed in Section IV is here used. Based on previous evaluations, the parameters used for running the observation platform optimization algorithm are $c = 1$, $c_1 = 200$, $c_2 = 200$, $ct^* = 90$, $\beta^{1*} = 0.08$, and $\beta^{2*} = 0.1$. Note that, typically, $\beta^1(O)$ and $\beta^2(O)$ are much smaller than the costs in Table II, and, hence, c_1 and c_2 have to be much larger than c for all terms in the loss index (5) comparable. A possible (local) optimal observation platform found by using the optimization algorithm described in Section IV consists of the following sensors:

$$s_{15}, s_{16}, s_{22}, s_{41}, s_{31}, s_{34}, s_{51}, s_{52}. \qquad (19)$$

Fig. 4 shows the sensor configuration in (19), where sensors selected (and not used) are crossed out. Note that sensors s_{15}, s_{16}, s_{22}, and s_{41} are used for detecting and counting

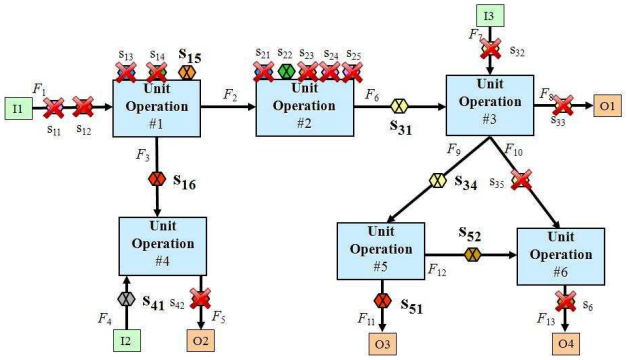

Fig. 4. Computed (local) optimal sensor configuration

the occurrences of f_1, while s_{31}, s_{34}, s_{51}, and s_{52} are used for f_2. For brevity, the observation platform consisting of the sensors in (19) is referred to as Observation platform (19) hereafter. Simulations of the monitored system using the baseline observation platform (which includes all available sensors) and Observation platform (19) were conducted using appropriately chosen false alarm tolerances. Fig. 5 plots the true and estimated (for both observation platforms simulated) numbers of occurrences of f_1 against the number of event executions, while Fig. 6 plots those for f_2. In both figures,

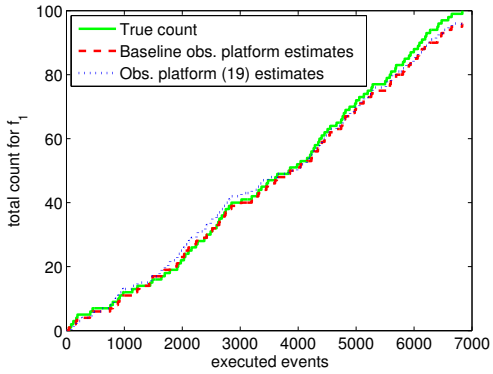

Fig. 5. Estimated vs true count of f_1 as function of executed events

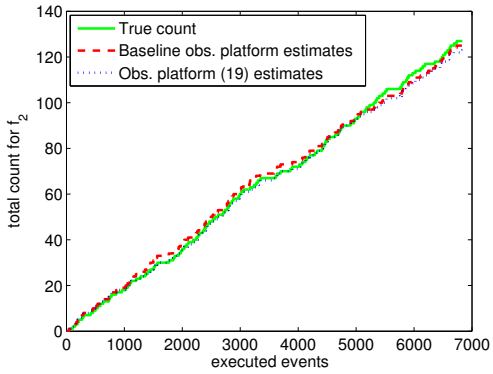

Fig. 6. Estimated vs true count of f_2 as function of executed events

estimate counts from the baseline observation platform and

Observation platform (19) follow closely the true anomaly occurrence counts at each and throughout a large number of executed events. Table III compares the cost and performance between these observation platforms, assuming α_i's that correspondingly minimize their ANEs. From Figs. 5 and 6 and Table III, Observation platform (19) achieves cost savings of $\frac{121-49}{121} = 59.5\%$, while the errors in estimating the occurrences of f_1 and f_2 are still reasonably small. The proposed approach for observation platform optimization scales well in practice. For example, for the actual facility considered here consisting of 9504 states, the algorithm takes about 18 hours to compute off-line a solution in a 64bit computer with Intel Xeon CPU E5520 running at 2.27 GHz.

TABLE III
COMPARISON BETWEEN OBSERVATION PLATFORMS

	Baseline obs. platform	Obs. platform (19)
Cost	121	49
$ANE^1(S)$	0.0437	0.0861
$ANE^2(S)$	0.0635	0.0671

VI. CONCLUSIONS

This paper considered the monitoring architecture illustrated in Fig. 1, where the monitored plant is modeled as a stochastic automaton, the sensors are unreliable, and a diagnoser is used to analyze observations. The sensors and the diagnoser together are referred to as the observation platform. A design methodology is proposed to find an observation platform that best balances cost and the performance of the diagnoser, while satisfying given observability requirements and constraints. This methodology adopts a simulation-based optimization algorithm that uses two greedy heuristics, one myopic and one that considers the projected performance of candidate sensors. These heuristics are sequentially executed in order to find improved observation platforms. The proposed methodology can be used to find (local) optimal observation platforms, although these solutions are not necessarily the global optimum. The proposed methodology was applied to an observation platform optimization problem for monitoring a multi-unit-operation system. The results showed that an optimal/improved observation platform can be found that may significantly reduce its cost, while still yielding acceptable performance for counting anomaly pattern occurrences.

ACKNOWLEDGEMENT

The research reported in this paper was supported by the U.S. Department of Energy contract DE-AC07-05ID14517.

REFERENCES

[1] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete event systems," *IEEE Trans. on Auto. Contr.*, vol. 40, no. 9, pp. 1555–1575, Sept 1995.

[2] T. Yoo and H. E. Garcia, "Diagnosis of behaviors of interest in partially-observed discrete-event systems," *Systems & Control Letters*, vol. 57, no. 12, pp. 1023–1029, 2008.

[3] J. Lunze and J. Schrder, "State observation and diagnosis of discrete-event systems described by stochastic automata," *Discrete Event Dyna. Syst.: Theory Appl.*, vol. 11, no. 4, pp. 319–369, 2001.

[4] D. Thorsley and D. Teneketzis, "Diagnosability of stochastic discrete-event systems," *IEEE Trans. Autom. Control*, vol. 50, no. 4, pp. 476–492, 2005.

[5] D. Throsley, T. Yoo, and H. E. Garcia, "Diagnosability of stochastic discrete event systems under unreliable observations," in *Proceedings of the American Control Conference*, 2008, pp. 1158 – 1165.

[6] E. Athanasopoulou, L. Lingxi, and C. Hadjicostis, "Probabilistic failure diagnosis in finite state machines under unreliable observations," in *Proc. of 2006 8th International Workshop on Discrete Event Systems*, Ann Arbor, MI, July 2006, pp. 301 – 306.

[7] ——, "Maximum likelihood failure diagnosis in finite state machines under unreliable observations," *IEEE Tran. on Autom. Control*, vol. 55, no. 3, pp. 579–593, 2010.

[8] L. K. Carvalho, J. C. Basilio, and M. V. Moreira, "Robust diagnosability of discrete event systems subject to intermittent sensor failures," in *Proc. of 10th International Workshop on Discrete Event Systems*, Berlin, Germany, Aug. 2010, pp. 94–99.

[9] S. T. S. Lima, J. C. Basilio, S. Lafortune, and M. V. Moreira, "Robust diagnosis of discrete-event systems subject to permanent sensor failures," in *Proc. of 10th International Workshop on Discrete Event Systems*, Berlin, Germany, Aug. 2010, pp. 100–107.

[10] O. Contant, S. Lafortune, and D. Teneketzis, "Diagnosis of intermittent faults," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 14, no. 2, pp. 171–202, 2004.

[11] S. Jiang, R. Kumar, and H. E. Garcia, "Diagnosis of repeated/intermittent failures in discrete-event systems," *IEEE Trans. Robotics and Applications*, vol. 19, no. 2, pp. 310–323, 2003.

[12] H. E. Garcia and T. Yoo, "Model-based detection of routing events in discrete flow networks," *Automatica*, vol. 4, no. 41, pp. 583–594, 2005.

[13] T. Yoo and H. E. Garcia, "New results on discrete-event counting under reliable and unreliable observation information," in *Proc. of 2005 IEEE Int. Conf. on Networking, Sensing and Control*, 2005.

[14] ——, "Event counting of partially-observed discrete-event systems with uniformly and nonuniformly bounded diagnosis delays," *Discrete Event Dynamic Systems*, vol. 19, no. 2, pp. 167–187, 2009.

[15] ——, "Stochastic event counter for discrete-event systems under unreliable observations," in *Proceedings of the American Control Conference*, 2008, pp. 1145 – 1152.

[16] W.-C. Lin, H. E. Garcia, D. Thorsley, and T. Yoo, "Sequential window diagnoser for discrete-event systems under unreliable observations," in *Proc. of 47th Annual Allerton Conf. on Communication, Control, and Computing*, Monticello, IL, Sept. 2009, pp. 668–675.

[17] T. Yoo, D. Thorsley, W.-C. Lin, and H. E. Garcia, "Recursive stochastic event counting for discrete-event systems under unreliable observations," submitted to *IEEE Tran. on Autom. Control*, 2009.

[18] T. Yoo and S. Lafortune, "NP-completeness of sensor selection problems arising in partially-observed discrete-event systems," *IEEE Trans. Automat. Contr.*, vol. 47, no. 9, pp. 1495–1499, 2002.

[19] T. Yoo and H. E. Garcia, "Sensor deployment optimization for network intrusion detection," in *Proc. of 44th Annual Allerton Conf. on Communication, Control, and Computing*, Monticello, IL, Sept. 2006, pp. 996–1004.

[20] W.-C. Lin, T. Yoo, and H. E. Garcia, "Sensor configuration selection for discrete-event systems under unreliable observations," in *Proc. of the 6th IEEE Conference on Automation Science and Engineering*, Toronto, Canada, Aug. 2010, pp. 477–484.

[21] L. Aguirre-Salas, O. Begovich, and A. Remírez-Treviño, "Optimal sensor choice for observability in free-choice Petri nets," in *Proc. of IEEE Int. Symposium on Intelligent Control*, Mexico City, Mexico, Sept. 2001, pp. 270–275.

[22] L. Aguirre-Salas, "Sensor selection for observability in interpreted Petri nets: a generic approach," in *Proc. of 42nd IEEE Conf. on Decision and Control*, Hawaii, Dec. 2003, pp. 3760–3765.

[23] S. Jiang, R. Kumar, and H. E. Garcia, "Optimal sensor selection for discrete event systems under partial observation," *IEEE Trans. Automat. Contr.*, vol. 48, no. 3, pp. 369–381, 2003.

[24] K. R. Rohloff, S. Kuller, and G. Kortsarz, "Approximating the minimal sensor selection for supervisory control," *Discrete Event Dyn. Syst.: Theory and Appl.*, vol. 16, no. 2, pp. 143–170, 2006.

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