Short Papers

Decision Fusion from Heterogeneous Sensors in Surveillance Sensor Systems

Elif I. Gokce, Abhishek K. Shrivastava, Jung Jin Cho, and Yu Ding, *Member, IEEE*

***Abstract*—Using multiple, heterogeneous sensors in surveillance systems is desirable, not only to tolerate sensor failures, but also to increase the accuracy of the event detection process and to provide complementary capability under different operating conditions. In the operation of multiple, heterogeneous sensors, we may encounter inconsistent sensor observations. Motivated by the need to make coherent decisions, we propose in this study a decision scheme to determine the right interpretations of sensor outputs when conflict arises. The proposed decision rule considers sensor heterogeneity in a surveillance system, while attempting to minimize the expected misclassification cost. Case studies of the surveillance sensor system in a major U.S. port demonstrate that the proposed decision scheme achieves a better robustness in the presence of sensor failures than the popular k-out-of-n decision fusion rule.**

***Note to Practitioners*—The central message from this paper is that when using heterogeneous sensor systems, the commonly used k-out-of-n decision fusion rule is no longer optimal. This has important implications for security surveillance applications because a complex surveillance sensor system almost always comprises multiple units of different types of sensors. Under those circumstances, a more flexible format of decision fusion should be allowed. Our paper presents a new optimal decision rule and a procedure to determine it. Information required for using this proposed decision rule includes the cost factors related to sensors and consequences of making wrong decisions, as well as the prior belief of how likely an adverse event would take place.**

***Index Terms*—Homeland security, robust decision fusion, sensor fusion, surveillance system for ports and waterways.**

I. Introduction

RECENT technological advances have enabled the development of surveillance sensor systems capable of gathering data from a specific region, processing the data, and making decisions based on the observed data. In this paper, we consider surveillance sensor systems designed to monitor restrictive security areas in ports and waterways. Fig. 1 shows an example of a surveillance sensor system for the Houston Ship Channel. In the ship channel, sensors continuously monitor specific regions called *surveillance points*. In order to increase the accuracy of surveillance and to provide complementary capabilities under different environmental conditions, each surveillance point is monitored by several sensors of different types.

When an event occurs at a surveillance point, the activity is picked up by each sensor monitoring the region. These sensor observations can be integrated to obtain decision about the event by using two different methods, namely data fusion [1] or decision fusion [2], [3]. In data fusion, each sensor sends its original measurement to the fusion center, which calculates a new estimate of the physical quantities under surveillance and makes a conclusion based on the new estimate. In decision fusion, each sensor sends a local decision (usually a binary one) derived by independent processing of its measurement. In surveillance sensor systems, since the outputs of the sensors is typically images or videos, the data fusion approach is usually difficult to implement. Therefore, the decision fusion approach is used, where local decisions are made either by an automated algorithm or a human operator. In this study we assume that binary local decisions are available at each sensor as "event" or "no-event."

In practice, when an event occurs at a surveillance point, not all sensors monitoring the region may report an "event." This happens because different sensors have different capabilities and their performance depends on environmental conditions. Moreover, sensors may fail or malfunction due to degradation or intentional tampering. This raises the question of how the final decision should be made, from individual decisions, in the presence of inconsistent sensor observations. For instance, what the final decision will be if two out of four sensors report "event" and the other two report "no-event"? If there is indeed no event at the point, but the sensor observations are interpreted as "event," then a *false alarm* occurs. Occurrence of false alarms will increase the operating cost as security forces will be sent to the respective area for investigation and interdiction. If an event actually takes place at the point, but the sensor outputs are interpreted as "no-event," then a *misdetection* occurs. Misdetection cost can be much higher than the false alarm cost because failure to detect and stop adversary activities in time may lead to catastrophic consequences. This paper addresses explicitly the manner in which individual decisions should be combined at a fusion center.

In this paper, we present a decision rule to optimally fuse binary sensor decisions in a surveillance sensor system. To the best of our knowledge, optimal decision rules proposed in literature (for example, [4]–[6]) are in the form of the k-out-of-n rule. That is, if at least k out of n sensors, monitoring a surveillance point, report "event," then the final decision is deemed as "event." This structure has been proven to minimize the expected misclassification cost (EMC) when all the sensors are identical [5]. But in our study, we consider surveillance sensor systems with heterogeneous sensors. Our study shows that for a heterogenous sensor system, the k-out-of-n structure is too rigid to always produce the best result in decision fusion. Our approach is to solve for the optimal decision rule, which minimizes the EMC, through a binary-programming formulation without imposing the k-out-of-n structure. We also present a numerical procedure that can efficiently compute the decision fusion results. The proposed decision rule is of more flexible format, and can better handle the situations when sensor failures occur. We will show that it is more fault-tolerant than the k-out-of-n decision fusion rule.

Manuscript received September 20, 2009; revised March 29, 2010; accepted July 24, 2010. Date of publication September 13, 2010; date of current version January 07, 2011. This paper was recommended for publication by Associate Editor R. Fierro and Editor S. Sarma upon evaluation of the reviewers' comments. This work was supported in part by the National Science Foundation under Grant CMMI-0529026 and Grant CMMI-0727305.

E. I. Gokce is with the Columbus Radiology Corporation, Columbus, OH 43215 USA (e-mail: elifilke@gmail.com).

A. K. Shrivastava is with the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Kowloon, Hong Kong (e-mail: Abhishek.Shrivastava@cityu.edu.hk).

J. J. Cho is with Baker Hughes Inc., Houston, TX 77019-2118 USA (e-mail: cho.jungjin@gmail.com).

Y. Ding is with the Department of Industrial and Systems Engineering, Texas A&M University, College Station, TX 77843-3131 USA (e-mail: yuding@iemail.tamu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2010.2064305

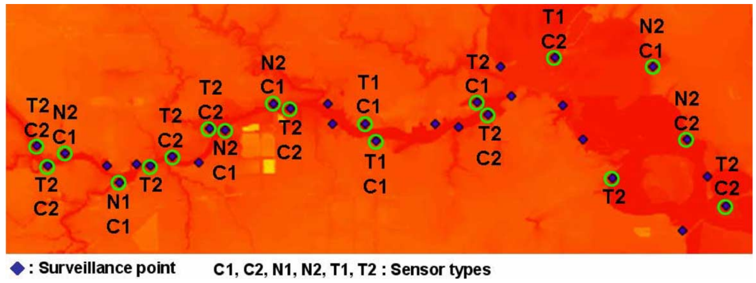

Fig. 1. A surveillance sensor system for the Houston Ship Channel. Letters "C", "T", and "N" stand for the three classes of surveillance sensors, i.e., CCTV, thermal/infrared, and night vision sensors, respectively. The number following the letters indicates the existence of multiple sensor types in a given sensor class. For example, C1 and C2 represent two different types of CCTV cameras.

The remainder of this paper is organized as follows. Section II reviews the related literature. Section III formulates the optimal decision fusion problem as a 0–1 integer program. Section IV describes the optimal decision rule that minimizes the expected misclassification cost and compares it with the commonly used k-out-of-n rule. We also present an algorithm to determine the proposed decision rule efficiently. Section V presents the case studies to computationally evaluate the performance of the proposed decision rule and compare it with the k-out-of-n rule. Finally, we conclude this paper in Section VI.

II. Relevant Literature

In this section, we briefly review the literature on decision fusion, which is the focus of the paper. For the related topic of data fusion, interested readers may refer to [7]–[10].

Classical theory of decision fusion is based on hypothesis testing and estimation methods [2], [3], [11], [12]. An earlier study [13] introduces a model under binary hypothesis H_1 (presence of an event) versus H_0 (absence of an event) and provides a theoretical framework for event detection using multiple sensors. Each sensor in the system makes an observation conditioned on the unknown hypothesis. Overall performance of the system is measured by the misdetection probability. If the sensor observations are independent, then the optimal decision at each sensor is made by employing a likelihood ratio threshold test with different threshold values. Since thresholds can be chosen in a number of ways and calculating them is generally difficult, most of the studies have focused on the cases where the same likelihood threshold is used at all sensors [1]. In that case, an optimal decision rule is a k-out-of-n rule [4]. Using Bayesian and Neyman–Pearson criteria, Zhang *et al.* [5] determine the optimal k and the likelihood threshold that minimizes the misdetection probability. Methods based on hypothesis testing are different from our study because, in the above-mentioned literature, a k-out-of-n rule structure is assumed, and the studies aim to find the optimal k and the optimal threshold at individual sensors.

A seminal work on decision fusion is by [6]. Based on the observation that the sensor faults are likely to be stochastically uncorrelated, while event measurements are likely to be spatially correlated, [6] proposes letting an individual sensor communicate with its neighbors and using other's decisions to correct its own decision. Assuming that the false alarm probability of a sensor is equal to its misdetection probability, [6] shows that the majority voting rule is the optimal decision rule that minimizes the average number of faulty observations. However, the majority voting rule does not hold its optimality when the assumption that the false alarm probability and misdetection probability are equal does not hold.

An alternative type of decision fusion is sequential decision making. In this, sensors sequentially send their observations to the fusion center, where a binary decision is made at a stopping time [14]. When to stop taking observations is a part of the overall decision procedure. Such a decision process has been used, for instance, for container inspection at the port-of-entry [15], [16]. The difference from our study is that the surveillance systems of interest here require fusing all observations simultaneously rather than sequentially.

Decision fusion utilizes processed information as opposed to the original measurements. Thus, one might expect that systems based on data fusion perform better than those based on decision fusion. However, [17] showed that data fusion has the same behavior as decision fusion under certain regularity conditions. Moreover, [18] compared algorithms based on data fusion and decision fusion. In the presence of faulty sensors in a system, [18] found that decision fusion-based algorithms may perform better than data fusion-based algorithms.

III. Misclassification Cost and Problem Formulation

Suppose a set of sensors I are deployed to observe a set of surveillance points J. Each surveillance point $j \in J$ is observed by a subset of sensors $I_j \subseteq I$ and each sensor $i \in I$ may observe more than one surveillance point. We assume that the performance of a sensor is not influenced by any other sensor in I_j and a sensor makes each of its decisions independently.

Let the actual occurrence of events at surveillance point j be denoted by the random variable U_j; i.e.,

$$U_j = \begin{cases} 1, & \text{if a suspicious event occurs at point } j \\ 0, & \text{otherwise.} \end{cases}$$

$P(U_j = 1)$ is the prior probability of a suspicious event occurring at j. Let Y_{ij} be the binary decision of sensor i for surveillance point j; i.e.,

$$Y_{ij} = \begin{cases} 1, & \text{if sensor } i \text{ reports "event" at point } j \\ 0, & \text{otherwise.} \end{cases}$$

Then, the misdetection probability of an event at point j by sensor i is $P(Y_{ij} = 0|U_j = 1)$ and, likewise, the false alarm probability is $P(Y_{ij} = 1|U_j = 0)$.

Let $\mathbf{y}_j = (y_{1j}, y_{2j}, \ldots, y_{n_j j})$ be the vector of binary 0–1 decisions from each of the $n_j (= |I_j|)$ sensors observing point j. Then, we denote the fused decision at j by $x(\mathbf{y}_j)$; i.e.,

$$x(\mathbf{y}_j) = \begin{cases} 1, & \text{if our fused decision at } j \text{ is "event"} \\ 0, & \text{otherwise.} \end{cases}$$

Let PFA_j and PMD_j be the false alarm and misdetection probabilities, respectively, and c_j^f and c_j^m be the false alarm and misdetection costs, respectively, at point j. Further, define D_E to be the set of all $\mathbf{y}_j$'s for which the fused decision is "event"; i.e., $D_E = \{\mathbf{y}_j : \mathbf{y}_j \in \{0,1\}^{n_j} \text{ s.t. } x(\mathbf{y}_j) = 1\}$, and D_{NE} to be the set of all $\mathbf{y}_j$'s for which the fused decision is "no-event"; i.e., $D_{NE} = \{\mathbf{y}_j : \mathbf{y}_j \in \{0,1\}^{n_j} \text{ s.t. } x(\mathbf{y}_j) = 0\}$. Then, the expected misclassification cost (EMC) at j is given by

$$C_j = c_j^f PFA_j + c_j^m PMD_j \tag{1}$$

where

$$PFA_j = \sum_{\mathbf{y}_j \in D_E} P(x(\mathbf{y}_j) = 1 | U_j = 0) P(U_j = 0)$$

and

$$PMD_j = \sum_{\mathbf{y}_j \in D_{NE}} P(x(\mathbf{y}_j) = 0 | U_j = 1) P(U_j = 1).$$

If $c_j^f = c_j^m = 1$, then $C_j = PMC_j$, the misclassification probability of the fused decision at j.

Observe in (1) that, for given $\mathbf{y}_j$, false alarm cost will be incurred only if $x(\mathbf{y}_j) = 1$; similarly, misdetection cost will be incurred only if $x(\mathbf{y}_j) = 0$. As such, EMC is

$$C_j = \sum_{\mathbf{y}_j \in \{0,1\}^{n_j}} \Big[c_j^f P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 0) P(U_j = 0) x(\mathbf{y}_j) + c_j^m P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 1) P(U_j = 1)(1 - x(\mathbf{y}_j)) \Big]. \quad (2)$$

Using (2), the problem of determining the optimal decision rule, which minimizes the EMC of sensors observing surveillance point j, becomes a binary integer programming problem

$$\begin{aligned} C_j^* = \min \sum_{\mathbf{y}_j \in \{0,1\}^{n_j}} \Big[& c_j^f P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 0) \times P(U_j = 0) x(\mathbf{y}_j) \\ & + c_j^m P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 1) \\ & \times P(U_j = 1)(1 - x(\mathbf{y}_j)) \Big] \\ \text{s.t.} \quad & x(\mathbf{y}_j) \in \{0,1\} \quad \text{for all} \quad \mathbf{y}_j \in \{0,1\}^{n_j}. \end{aligned} \quad (3)$$

It may be noted that, from a Bayesian decision theory perspective, C_j is the Bayes risk of $x(\mathbf{y}_j)$ using the 0/1 loss function and $x^*(\mathbf{y}_j)$ is the Bayes estimator.

IV. Decision Rule

A. Optimal Decision Rule (ODR)

The following theorem gives an explicit description of the ODR, the optimal solution to Problem (3). (To save space, proofs of the results presented here have been omitted. They are available at the corresponding author's website: http://ise.tamu.edu/metrology.)

Theorem 1: The optimal solution x^* to Problem (3) is given by equation (4) at the bottom of the page.

While Theorem 1 is easy to understand, it is not convenient to use in practice. In order to present it in a practically useful format, we conduct the following analysis. Let

$$T(\mathbf{y}_j) = \frac{P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 1)}{P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 0)} \quad \text{and} \quad t_j = \frac{c_j^f P(U_j = 0)}{c_j^m P(U_j = 1)}.$$

Then, according to Theorem 1, $\mathbf{y}_j$ is interpreted as "event" (i.e., $x^*(\mathbf{y}_j) = 1$) if

$$T(\mathbf{y}_j) \geq t_j \quad (5)$$

and as "no-event" (i.e, $x^*(\mathbf{y}_j) = 0$) otherwise (i.e., $T(\mathbf{y}_j) < t_j$). Now, since the sensors are independent, we have

$$\begin{aligned} T(\mathbf{y}_j) &= \frac{\prod_{i \in I_j} P(Y_{ij} = y_{ij} | U_j = 1)}{\prod_{i \in I_j} P(Y_{ij} = y_{ij} | U_j = 0)} \\ &= \prod_{i \in I_j} \frac{P(Y_{ij} = y_{ij} | U_j = 1)}{P(Y_{ij} = y_{ij} | U_j = 0)}. \end{aligned}$$

Let $\mathbf{y}_j^* = \arg\max \{T(\mathbf{y}_j) | \mathbf{y}_j \in \{0,1\}^{n_j}\}$ and $T_{\max} = T(\mathbf{y}_j^*)$. Assuming that the sensors observing j provide acceptable performance in the absence of any failures we clearly have $x^*(\mathbf{y}_j^*) = 1$. Note that $\mathbf{y}_j^*$ can be determined by simply maximizing the contributions of each sensor i in $T(\mathbf{y}_j^*)$ separately, which reduces to setting $y_{ij}^* = 1$ if $(P(Y_{ij} = 1 | U_j = 1)/P(Y_{ij} = 1 | U_j = 0)) \geq (P(Y_{ij} = 0 | U_j = 1)/P(Y_{ij} = 0 | U_j = 0))$ and $y_{ij}^* = 0$, otherwise.

Using $\mathbf{y}_j^*$ and $T_{\max}$, Lemma 1 presents a different expression for $T(\mathbf{y}_j)$.

Lemma 1:

$$T(\mathbf{y}_j) = T_{\max} \prod_{i \in S(\mathbf{y}_j)} \frac{1}{\delta_i}$$

where

$$\delta_i = \frac{P\left(Y_{ij} = y_{ij}^* | U_j = 1\right)}{P\left(Y_{ij} = y_{ij}^* | U_j = 0\right)} \times \frac{P\left(Y_{ij} = 1 - y_{ij}^* | U_j = 0\right)}{P\left(Y_{ij} = 1 - y_{ij}^* | U_j = 1\right)}$$

and

$$S(\mathbf{y}_j) = \left\{ i \in I_j | y_{ij} \neq y_{ij}^* \right\}.$$

Using Lemma 1 and (5), we have the following simplification of the ODR.

Lemma 2: Let $\beta = T_{\max}/t_j$. In ODR, all outputs $\mathbf{y}_j \in \{0,1\}^{n_j}$ such that

$$\prod_{i \in S(\mathbf{y}_j)} \delta_i \leq \beta \quad (6)$$

are interpreted as "event"; all the remaining outputs are interpreted as "no-event."

Lemma 2 provides a convenient format of the ODR as only the δ_i's depend on sensor outputs whereas β is fixed once the surveillance system is known. To understand the ODR specified in Lemma 2, consider the following. $T(\mathbf{y}_j)$ is the likelihood ratio, given a sensor output $\mathbf{y}_j$, that an event occurs versus no-event, and $T_{\max}$ is the maximum likelihood ratio, representing the best potential of the sensor system in detecting an event. The quantity β on the right-hand side does not depend on the actual sensor outputs; rather it depends on the maximum likelihood ratio, the misclassification cost and the event's prior probability ratios (deciding t_j). In this sense, β can be considered as a performance threshold determined by the sensor system and the surveillance task. On the left-hand side, δ_i represents the capability of sensor i (i.e., false alarm rate and detection power). Apparently, the ODR specified in Lemma 2 is decided by comparing the sensor detection capability with a system-level threshold. It is also worth noting that Lemma 2 can

$$x^*(\mathbf{y}_j) = \begin{cases} 1, & \text{if } c_j^m P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 1) P(U_j = 1) \geq c_j^f P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 0) P(U_j = 0) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

TABLE I
SENSOR INFORMATION FOR FOUR SENSOR EXAMPLE

i	$P(y_{ij}=1 \mid U_j=1)$	$P(y_{ij}=1 \mid U_j=0)$	$P(y_{ij}=0 \mid U_j=1)$	$P(y_{ij}=0 \mid U_j=0)$	δ_i
1	0.90	0.10	0.10	0.90	81.00
2	0.60	0.45	0.40	0.55	1.83
3	0.99	0.01	0.01	0.99	9801.01
4	0.90	0.04	0.10	0.96	216.00

ODA (I_j)

Step 0: Determine δ_i for $i \in I_j$, $\mathbf{y}_j^*$, and β

Step 1: Apply ProcQ(I_j, β) to generate Q

Step 2: For each $S \in Q$

Set $y_{ij} = 1 - y_{ij}^*$ for $i \in S$

$y_{ij} = y_{ij}^*$ for $i \in I_j \setminus S$

Report that $\mathbf{y}_j$ means "event"

Fig. 2. Optimal decision algorithm.

ProcQ (I_j, β)

Step 0: Set $t = 1$, $r = \text{argmax}\{i \in I_j \mid \delta_i \leq \beta\}$,

$S_i^t = \{i\}$ for $i = \{1, \ldots, r\}$,

$Q = \{S_1^t, \ldots, S_r^t\}$

Step 1: For each $S_\ell^t \in Q$ such that $\ell < r$,

$q = \text{argmax}\{i = 1, \ldots, r \mid \prod_{z \in S_\ell^t} \delta_z \leq \frac{\beta}{\delta_i}\}$

for $i = \ell + 1, \ldots, q$

Set $S_i^{t+1} = S_\ell^t \cup \{i\}$ and $Q = Q \cup S_i^{t+1}$

end for

Step 2: Set $t = t + 1$

If $t < r$, then go to Step 1

else Stop

Fig. 3. Procedure ProcQ.

be used not only for individual sensors, but also for sensor combinations (pay attention to the product at the left-hand side over the set of sensors in S).

B. ODR Versus k-out-of-n Rule

To realize that the ODR is different from a k-out-of-n rule, consider an example of four sensors monitoring a single surveillance point, with sensor characteristics as given in Table I. Let $P(U_j = 1) = 0.02$, $c_j^f = 100$, and $c_j^m = 250$. Then, $\mathbf{y}_j^* = (1, 1, 1, 1)$ and $\beta = 1{,}360.37$. The last column in Table I gives the δ_i values for the sensors. For $\mathbf{y}_j = (1, 1, 1, 0)$ and $\mathbf{y}_j = (1, 1, 0, 1)$, the decisions according to ODR are 1 (event) and 0 (no-event), respectively. This clearly is in conflict with any k-out-of-n rule.

While the ODR is different from the k-out-of-n rule in general, the ODR reduces to a k-out-of-n rule when the sensors are identical. In that case, we have $\delta_1 = \delta_2 = \cdots = \delta_{n_j} = \delta$ and $\mathbf{y}_j^* = (1, 1, \ldots, 1)$. Thus, Lemma 2 implies, for all $S \subseteq I_j$ such that $\delta^{|S|} \leq \beta$, the output $\mathbf{y}_j$, where $y_{ij} = 1$ for $i \in I_j \setminus S$ and $y_{ij} = 0$ for $i \in S$,

TABLE II
TEST INSTANCES [19]

N	$\lvert T \rvert$	$\lvert J \rvert$	n	min n_j	ave. n_j	max n_j
1	1	42	23	2	4	6
2	1	42	26	2	4	6
3	1	42	22	2	4	6
4	1	42	26	2	4	7
5	3	42	25	2	4	6
6	3	42	23	2	4	6
7	3	42	22	2	4	6
8	3	42	24	2	4	6
9	1	84	38	2	3	5
10	1	84	40	2	3	5
11	1	84	37	2	3	5
12	1	84	40	2	3	4
13	3	84	41	2	4	6
14	3	84	41	2	4	5
15	3	84	40	2	3	6
16	3	84	39	2	3	5

means "event." In other words, if $(n_j - \lfloor ln(\beta)/ln(\delta) \rfloor)$ sensors report "event," then the decision from our ODR is "event," effectively a $(n_j - \lfloor ln(\beta)/ln(\delta) \rfloor)$-out-of-$n_j$ rule.

C. Optimal Decision Algorithm

The ODR proposed in Section IV-A can be generated (as a list of $\mathbf{y}_j$'s for which the decision is "event") by exhaustively enumerating all possible sensor outputs ($\mathbf{y}_j$'s) and using the condition in Lemma 2. But this may be computationally demanding for a large surveillance system. In this section, we present an efficient algorithm to achieve this task. The algorithm primarily attempts to exploit the properties of the δ_i's to avoid complete enumeration of sets.

Fig. 2 shows the overall algorithm for generating the ODR for a surveillance point j. Step 1 is the main step of the algorithm, which calls the procedure ProcQ. The procedure ProcQ, detailed in Fig. 3, efficiently finds the set Q, consisting of all subsets S of I_j such that $\prod_{i \in S} \delta_i \leq \beta$.

TABLE III
DESCRIPTION OF DECISION RULES

Rule name	**Conditions under which the rule reports "event"**
1-out-of-n	One of the sensors that observe j reports "event"
n-out-of-n	All sensors that observe j report "event"
Majority	More than half of the sensors that observe j report "event"
Best-k	For the sensors that observe j, let k_j^*-out-of-n rule be the k-out-of-n rule with the smallest EMC. At least k_j^* of the sensors that observe j report "event"

ProcQ assumes that the indices in I_j have been ordered such that $\delta_1 \leq \delta_2 \leq \cdots \leq \delta_{n_j}$. In Fig. 3, S_ℓ^m denotes a subset of I_j, where ℓ denotes the largest index in S_ℓ^m (i.e., $\ell \in \arg\max \{\delta_i | i \in S_\ell^m\}$) and m is the cardinality of S_ℓ^m (i.e., $m = |S_\ell^t|$). For example, $S_i^1 = \{i\}$; $S_4^2 = \{1, 4\}$. ProcQ starts with generating the single-element sets in Q. Then, at each iteration t, the $(t+1)$-element sets in Q are generated using the t-element sets. The process iterates until all sets in Q are generated.

The run time of computing all $\delta_1, \ldots, \delta_{n_j}$'s and their combinations by calculating all subsets is $O(2^{n_j})$. However, because we do not investigate all subsets in ProcQ, our run time is less than $O(2^{n_j})$. At each iteration t, ProcQ checks at most $(n_j - t)(n_j + 1 - t/2) < O\left(n_j^2\right)$ sets that are not in Q. Since there are at most n_j iterations, the run time of ProcQ is $O\left(|Q| + n_j^3\right)$.

Step 0 in ODA takes $O(n_j)$ time. Since Step 1 applies ProcQ to generate Q, its run time is $O\left(|Q| + n_j^3\right)$. The run time of Step 2 is $O(n_j|Q|)$. The dominant run time of ODA is Step 2. Thus, ODA runs in $O(n_j|Q|)$. If we used implicit enumeration method instead of ProcQ in ODA, the run time would be $O\left(n_j 2^{n_j}\right)$.

V. CASE STUDY AND EVALUATION

We use the surveillance sensor system in a major US port to compare the proposed ODR and the existing k-out-of-n decision rule. Wilhelm and Gokce [19] generated optimal surveillance system layouts for 16 different realistic instances of the Houston Ship Channel. We use these 16 instances (see Table II) in our study. Due to space limitation, we omit the detailed description of the instances, but provide some summary information in Table II; for detailed description please refer to [19]. In Table II, N denotes the instance number. Fig. 1 shows the layout of surveillance sensors for instance 16.

For each of the 16 test instances, we simulate the probabilistic detection behavior of the surveillance sensor system, including random sensor failures. We assume that a failed sensor always reports "no-event." We compare several variants of the k-out-of-n rule, listed in Table III, with the proposed ODR in these simulations. The best-k rule uses different k's for different surveillance points. For each surveillance point j, the k_j^* value is chosen by comparing the performance of every possible k-out-of-n rule. Each simulation is run for 10^9 event occurrences so that inferences are based on the steady-state behavior of the surveillance system.

In the simulation study, we use the misdetection probabilities (MDPs) and sensor failure probabilities given in [19]. However, [19] does not specify the false alarm probabilities (FAPs) or the misclassification costs (i.e., false alarm and misdetection costs). Since we did not find any consensus in literature for selecting the FAPs, we compared the decision rules for various choices of FAPs, both fixed and varying with the MDPs. The conclusions were similar. Here we only present the results for two cases: $\mathrm{FAP} = \mathrm{MDP}$ and $\mathrm{FAP} = \mathrm{MDP}/100$. For

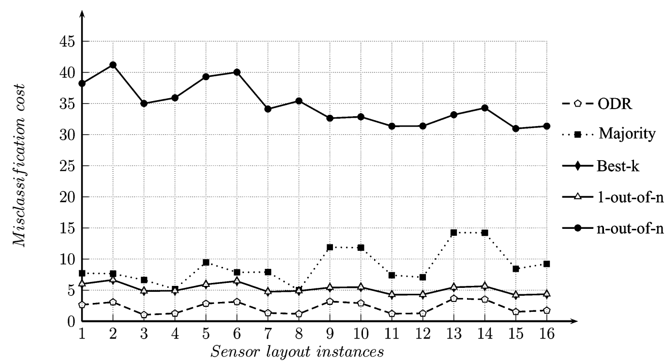

Fig. 4. $\mathrm{FAP} = \mathrm{MDP}, c_m = 100c_f$.

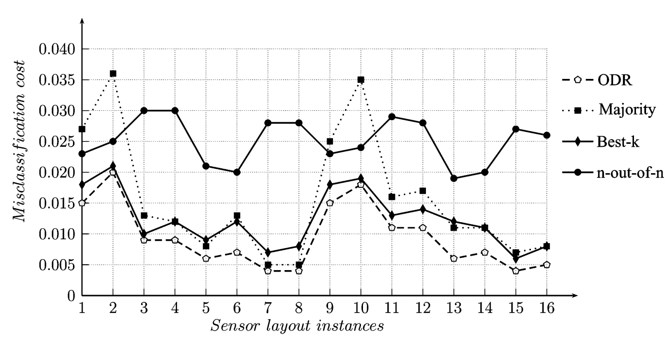

Fig. 5. $\mathrm{FAP} = \mathrm{MDP}/100, c_m = c_f$.

the misclassification costs, we again consider two cases: $c_j^m/c_j^f = 100$ and $c_j^m/c_j^f = 1$. We believe that the first case better reflects a surveillance application as the consequence of a missed detection can have a much more serious consequence than a false alarm.

The comparison results for $\mathrm{FAP} = \mathrm{MDP}, c_m = 100c_f$ and $\mathrm{FAP} = \mathrm{MDP}/100, c_m = c_f$ are presented in Figs. 4 and 5, respectively. To save space, we have omitted the results for the other two cases as they lead to the same conclusion. However, these are available at the corresponding author's website (http://ise.tamu.edu/metrology). Note that in Fig. 5, results for the 1-out-of-n are not shown. This is because the misclassification cost in this case is an order of magnitude larger than the other rules. From the figures, it can be observed that the proposed ODR outperforms all the k-out-of-n variants in all the scenarios. Since the decision rule computations (implicitly) assume that all the sensors are working, these results clearly show that the proposed ODR is more robust than any variant of the k-out-of-n rule. Further, among the different k-out-of-n rules, we find that there is no single rule that outperforms the others in all the cases (for instance see Fig. 5). While the best-k rule is optimal, it is so only when there are no sensor failures.

VI. Conclusion

In this paper, we propose a decision fusion rule to determine the robust interpretations of outputs from heterogeneous surveillance sensors. We formulate the event detection problem as a 0–1 integer program. Using this model, we define the optimal decision rule that minimizes the expected misclassification cost. Using 16 test instances of a surveillance sensor system for a major U.S. port, we compare the performance of the proposed ODR with several variants of the popular k-out-of-n rule. Our comparisons demonstrate that the proposed ODR is more robust to sensor failures, always attaining the smallest misclassification cost. Intuitively, the robustness is because the ODR does not depend on the number of sensors (it mostly disregards any number of poor performing sensors) whereas the k-out-of-n rule does.

An important extension to the problem considered here is the case of sequential decision making from heterogeneous sensors. This is relevant to scenarios where a target object may be observed over a period of time before taking a decision. One way ODR may be extended to this case is by using event prior probabilities updated with the ODR performance at the previous decision point. However, this idea needs to be refined and forms part of our future work.

References

[1] P. K. Varshney, *Distributed Detection and Data Fusion*. New York: Springer-Verlag, 1997.

[2] R. Viswanathan and P. K. Varshney, "Distributed detection with multiple sensors: Part 1—Fundamentals," *Proc. IEEE*, vol. 85, pp. 54–63, Jan. 1997.

[3] R. S. Blum, A. Kassam, and H. V. Poor, "Distributed detection with multiple sensors: Part 2—Advanced topics," *Proc. IEEE*, vol. 85, pp. 64–79, Jan. 1997.

[4] W. Shi, T. W. Sun, and R. D. Wesel, "Quasi-convexity and optimal binary fusion for distributed detection with identical sensors in generalized Gaussian noise," *IEEE Trans. Inform. Theory*, vol. 47, no. 1, pp. 446–450, Jan. 2001.

[5] Q. Zhang, P. K. Varshney, and R. D. Wesel, "Optimal bi-level quantization of i.i.d. sensor observations for binary hypothesis testing," *IEEE Trans. Inform. Theory*, vol. 48, no. 7, pp. 2105–2111, Jul. 2002.

[6] B. Krishnamachari and S. Iyengar, "Distributed bayesian algorithms for fault-tolerant event region detection in wireless sensor networks," *IEEE Trans. Comput.*, vol. 53, no. 3, pp. 241–250, Mar. 2004.

[7] K. Marzullo, "Tolerating failures of continuous-valued sensors," *ACM Trans. Comput. Syst.*, vol. 8, no. 4, pp. 284–304, 1990.

[8] L. Prasad, S. S. Iyangar, R. L. Rao, and R. L. Kashyap, "Fault-tolerant sensor integration using multiresolution decomposition," *Amer. Phys. Soc.*, vol. 49, no. 4, pp. 3452–3461, 1994.

[9] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proc. IEEE*, vol. 85, pp. 6–23, Jan. 1997.

[10] E. F. Nakamura, A. A. F. Laureiro, and A. C. Frery, "Information fusion for wireless sensor networks: Methods, models, and classifications," *ACM Comput. Surveys*, vol. 39, no. 3, pp. 1–55, 2007.

[11] S. C. A. Thomopoulos, R. Viswanathan, and D. K. Bougoulias, "Optimal distributed decision fusion," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 25, no. 5, pp. 761–765, Sep. 1989.

[12] X. Luo, M. Dong, and Y. Huang, "On distributed fault-tolerant detection in wireless sensor networks," *IEEE Trans. Comput.*, vol. 55, no. 1, pp. 58–70, Jan. 2006.

[13] R. R. Tenney and N. R. Sandell, "Detection with distributed sensors," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 17, no. 4, pp. 501–510, 1981.

[14] V. V. Veeravalli, "Sequential decision fusion: Theory and applications," *J. Franklin Inst.*, vol. 336, pp. 301–322, 1999.

[15] D. Madigan, S. Mittal, and F. Roberts, "Sequential decision making algorithms for port of entry inspection: Overcoming computational challenges," in *Proc. IEEE ISI*, New Brunswick, NJ, May 2007, pp. 1–7.

[16] E. A. Elsayed, C. M. Young, M. Xie, and Y. Zhu, "Port-of-entry inspection: Sensor deployment policy optimization," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 2, pp. 265–276, Apr. 2009.

[17] M. M. Kokar, J. A. Tomasik, and J. Weyman, "Data vs. decision fusion in the category theory framework," in *Proc. FUSION*, 2001, vol. 1, pp. TuA3-15–TuA3-20.

[18] T. Clouquer, K. K. Saluje, and P. Ramanathan, "Fault tolerance in collaborative sensor networks for target detection," *IEEE Trans. Comput.*, vol. 53, no. 3, pp. 320–333, Mar. 2004.

[19] W. E. Wilhelm and E. I. Gokce, "Branch-and-price decomposition to design a surveillance system for port and waterway security," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 2, pp. 316–325, Apr. 2010.

Distributed Optimization for Model Predictive Control of Linear Dynamic Networks With Control-Input and Output Constraints

Eduardo Camponogara and Helton F. Scherer

***Abstract*—A linear dynamic network is a system of subsystems that approximates the dynamic model of large, geographically distributed systems such as the power grid and traffic networks. A favorite technique to operate such networks is distributed model predictive control (DMPC), which advocates the distribution of decision-making while handling constraints in a systematic way. This paper contributes to the state-of-the-art of DMPC of linear dynamic networks in two ways. First, it extends a baseline model by introducing constraints on the output of the subsystems and by letting subsystem dynamics to depend on the state besides the control signals of the subsystems in the neighborhood. With these extensions, constraints on queue lengths and delayed dynamic effects can be modeled in traffic networks. Second, this paper develops a distributed interior-point algorithm for solving DMPC optimization problems with a network of agents, one for each subsystem, which is shown to converge to an optimal solution. In a traffic network, this distributed algorithm permits the subsystem of an intersection to be reconfigured by only coordinating with the subsystems in its vicinity.**

***Index Terms*—Convex optimization, distributed optimization, interior-point methods, linear systems, model predictive control.**

I. Introduction

Model predictive control (MPC) is a leading technology for controlling complex dynamic systems, mostly because of its ability to handle constraints systematically and the potential to reach optimal solutions [1]. The applications of MPC abound, including the control of logistics networks [2], mobile robots [3], and intelligent transportation systems [4]. In essence, MPC converts a dynamic control problem into a series of time-overlapping static optimization problems that are solved with standard optimization algorithms. At each sample time, the system state is measured and an optimization problem is solved over a finite-time horizon. Only the control signals for the first time interval are implemented, but the long-term effects on the objective are accounted for in the predictions.

The centralization of computation and the communication with remote sensors are the principal obstacles to the application of model

Manuscript received April 23, 2009; revised December 21, 2009; accepted July 04, 2010. Date of publication August 23, 2010; date of current version January 07, 2011. This paper was recommended for publication by Associate Editor P. Liu and Editor M. Zhou upon evaluation of the reviewers' comments. This was supported in part by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) under Grant 472439/2009-0.

The authors are with the Department of Automation and Systems Engineering, Federal University of Santa Catarina, Cx.P. 476, Florianópolis, SC 88040-900, Brazil (e-mail: camponog@das.ufsc.br, scherer@das.ufsc.br).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2010.2061842