Fault Tolerance Analysis of Surveillance Sensor Systems

Elif I. Gokce, Abhishek K. Shrivastava, and Yu Ding, *Senior Member, IEEE*

***Abstract*—A surveillance sensor system is a network of sensors that provides surveillance coverage to designated geographical areas. If all sensors are working properly, a well-designed surveillance system can supposedly provide the desirable level of detection capability for the locations and regions it covers. In reality, sensors may fail, falling out-of-service. Motivated by the need to determine the ability of a surveillance sensor system to tolerate the failure of sensors, we propose a fault tolerance capability measure to quantify the robustness of surveillance systems. The proposed measure is a conditional probability, characterizing the likelihood that a surveillance system is still working in the presence of sensor failures. Case studies of the surveillance sensor system in a major US port demonstrate that this new measure differentiates different surveillance systems better than using the sensor redundancy measure, or the reliability measure.**

***Index Terms*—Detection capability, multi-sensor combination, sensor fault, sensor network, surveillance for ports and waterways.**

ACRONYMS

AGP	art gallery problem
CCTV	closed-circuit television
FAP	false alarm probability
FTC	fault tolerance capability
HSC	Houston ship channel
MDP	misdetection probability
RL	redundancy level

NOTATION

I	set of sensors
n	size of set I
J	set of surveillance points
I_j	set of sensors monitoring surveillance point j
n_j	size of set I_j
J_i	set of surveillance points monitored by sensor i
X	set of working sensors, $X \subseteq I$
$I^{(X^-)}$	set of failed sensors, equals $I \setminus X$
c_j^m	cost of misdetections at surveillance point j
c_j^f	cost of false alarms at surveillance point j
$C_j(X)$	misclassification cost at surveillance point j, when monitored by sensors X
L_j	threshold misclassification cost for a working sensor system
δ_{ij}	measure of i^{th} sensor's capability at surveillance point j

Manuscript received March 11, 2012; revised November 26, 2012; accepted January 09, 2013. Date of publication May 13, 2013; date of current version May 29, 2013. This work was supported in part by the National Science Foundation under Grants CMMI-0529026 and CMMI-0727305, and by the City University of Hong Kong under Grant 7002706. Associate Editor: H. Li.

E. I. Gokce is with Bank of America, Dallas, TX 75202-3714 USA (e-mail: elifilke@gmail.com).

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

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/TR.2013.2259192

I. INTRODUCTION

We in this paper address the question of how to quantify the capability that a surveillance sensor system tolerates the failure of sensors. A surveillance sensor system is a network of sensors that provides surveillance coverage to designated geographical areas. Ideally, if all sensors are working properly, a well-designed surveillance system can supposedly provide the desirable level of detection capability for the locations and regions it covers. In reality, sensors may fail and become out-of-service. This raises the question how robust a surveillance sensor system remains in the presence of sensor failures.

We in this paper consider surveillance sensor systems designed to monitor restrictive security areas in ports and waterways. But the features of the system, and the research question we mean to address, extend to other surveillance sensor systems as well. In the sequel, we use the surveillance sensor system at the Houston ship channel (HSC), which has been studied in previous publications [1], [2], to explain the basics of such a system, and some related terminologies and symbolism. Refer to Fig. 1 of [2] for a graphical illustration of the HSC system.

The HSC surveillance system comprises multiple types of sensors, including closed-circuit television (CCTV), night vision enabled CCTV, infrared cameras, and radars. Heterogeneity among sensors commonly exists in surveillance systems, providing surveillance under very different lighting and ambient conditions (such as day versus night, presence

of precipitation or not). The set of sensors is denoted by $I := \{1, \ldots, n\}$.

The HSC system covers a long, curved strip along the ship channel. It is customary to designate a set of discrete points along this strip as the so-called *surveillance points*, at which sensors continuously monitor activities, and would trigger alarms when probable foul plays are detected. Choices of the surveillance points can be made by either the owner of the surveillance system (that is, the US Coast Guard), or in consultation with tenants occupying the area along the channel (such as Shell Oil). The set of surveillance points is denoted by J.

To provide surveillance coverage along the ship channel, all the sensors are either installed on a sensor tower or on top of a nearby building, ensuring an unobstructed line of sight between a sensor and a surveillance point. Multiple sensors of various types can be installed at the same location (on a single sensor tower, for example). One sensor can monitor multiple surveillance points. On the other hand, a surveillance point is usually monitored by more than one sensor. Denote by J_i the number of surveillance points that sensor $i \in I$ monitors, by I_j the subset of sensors monitoring surveillance point $j \in J$, and by n_j the number of sensors in I_j.

To get a real sense of the system composition, consider the sixteen layouts of the surveillance sensor system presented in [1]: the number of sensors can be as many as $n = 41$; three different sensor types are used; the number of surveillance points is either 42 (single bank), or 84 (both banks); n_j ranges from two to seven, and the average number of points a sensor monitors is about six.

Even when there is a clear line of sight between a sensor and a surveillance point, it does not guarantee that all events taking place at the surveillance point can always be detected and classified correctly. This detection function of a surveillance system is complicated, involving automated recognition algorithms or human operator's decisions or both. We here choose to consider the whole detection and surveillance process as a black box, and characterize its capability by a misclassification cost C, which combines the effects of both missed detections and false alarms. The misclassification cost for point $j \in J$ under the surveillance of sensor $i \in I$ is then given by $C_j(\{i\}) = c_j^m \cdot (1 - d_j(\{i\})) \cdot P(event) + c_j^f \cdot f_j(\{i\}) \cdot P(no\ event)$, where $P(event)$, and $P(no\ event)$ are the prior probabilities of the presence, and absence of any suspicious event at point j, respectively. In the case where people consider that misdetections greatly outweigh the risk of false alarms, implying that $c_j^m \gg c_j^f$, then $C_j(\{i\})$ deteriorates to simply the misdetection cost, whereas in the case where $c_j^m = c_j^f$, the misclassification cost becomes the commonly known misclassification error rate.

Understandably, the detection probability, as well as the false alarm probability, are affected by a number of factors, such as the distance between the sensor and the surveillance point, and can vary under different ambient conditions. Considering that both the sensor towers and the surveillance points, once the system is built, are stationary, and that we can select one fixed ambient condition (out of a finite number of such conditions) for our study, we believe it is justifiable to treat these probabilities between sensor $i \in I$ and point $j \in J$ as constants.

When we say a sensor is working, the sensor functions with designed detection and false alarm probabilities. When we say that a sensor has failed, it refers to the situation when the sensor is incapacitated, so that this sensor can be removed from the system. In the example of a CCTV, a failed sensor simply sends back all-white, all-black, static-noise images, or a stationary image not reflecting the activities happening at the surveillance point.

The system of surveillance sensors is considered working if all the surveillance points are provided surveillance with a misclassification cost lower than a prescribed threshold; otherwise, the system is considered failed. Based on this definition, it is not surprising to see that a surveillance sensor system may still be working when certain sensors have failed. The question we intend to answer is whether we can, and how to, use a simple measure to rank the capability of a surveillance sensor system in tolerating sensor failures (hereafter called *fault tolerance capability*, or FTC).

In the remainder of the paper, we first argue, through reviewing the existing work, that a new FTC measure is indeed needed for quantifying the robustness of surveillance systems. Following this, we will propose a FTC measure, and argue why this new measure is sensible to use. We then present structural results that allow this new measure to be computed efficiently. The next section presents a case study using the sixteen surveillance sensor layouts in [1], supporting the claims we make thus far. Finally, we end the paper with some concluding remarks.

II. Existing Work

The concept of fault tolerance is closely related to the concept of reliability [3]. Each sensor has its own reliability, i.e., the probability that a sensor will remain in the working condition. A sensor's reliability can be assessed by extensive testing (by the sensor's vendor), or pooling empirical data on those that had been in service under comparable conditions. Obviously, an individual sensor's reliability falls short of providing a comprehensive picture at the system level.

Reliability engineers specialize in aggregating all components' reliability to model a multi-component system. This activity can certainly be done for a network of sensors, which means to sort through different combinations of sensor failures, compute the system failure probability under each circumstance, and then take the weighted average of all plausible circumstances to yield a final probability measure as the system's reliability [4]–[6]. Here we argue that the FTC measure we conceive is not the same as reliability. Reliability is an unconditional probability, while FTC means to articulate the robustness of a system in states in which some sensors have failed. Intuitively speaking, system reliability is the combined effect of FTC and the individual sensors' reliability. Later in Section III, we will show an example where, although the sensors' prior reliability levels are all the same, the system's FTC can be different by changing how the sensors are deployed. The benefit of devising and using such a system-articulating FTC measure is to compare and select a robust design of the system.

Another relevant, and also relatively large, body of literature exists on sensor fusion, or more specifically, fault-tolerant

sensor fusion, for example, [2], [7]–[11], among others. This line of research is to devise robust procedures to combine observations or decisions from individual sensors in the possible presence of erratic sensor outputs. This type of work is also closely related to the field of robust statistics, where robust estimators are developed to provide estimations less sensitive to outliers [12], [13]. The difference between the fault-tolerant sensor fusion and the FTC measure can be understood as follows. Existing methods of sensor fusion do not provide system-level characteristics for cases where multiple surveillance points are observed by multiple sensors as they are most appropriate to characterize individual surveillance points where a surveillance point is observed by multiple sensors. Further, a sensor fusion can be done without necessarily knowing explicitly the fault-tolerance capability of the fusion procedure. In fact, many sensor fusion methods did not present a FTC analysis.

People did study, for some sensor fusion algorithms [14]–[16], how many anomalous sensors such a procedure can tolerate while still yielding the correct outcome. This measure manifests in the form of the *degree of sensor redundancy*. In robust statistics, the related robustness concept is called the *breakdown point* [17], which has been shown in [18] to be equivalent to the degree of redundancy. Depending on the form of system models, computing the degree of sensor redundancy invokes either graph theory, if the sensor network can be modeled as a graphic network [4], [14], [16], or matrix or matroid theory, if a robust estimation pertinent to a linear model is of concern [6], [15], [18], [19]. In both cases, the minimum cut of an appropriate algebraic structure (being a graph or a matrix) needs to be found. On this ground, this line of work is connected with reliability theory, where finding the minimum cut is a frequent exercise.

Applying the redundancy concept to a surveillance system is not new; modifying the art gallery problem (AGP) [20] to allow redundant guards is a complimentary problem. AGP seeks to find the minimum number of guards that have a direct line of sight to every point in a polygon-shaped art gallery, and the redundancy-allowing AGP is to increase the minimum number of guards allowed so that there may be multiple lines of sight to certain points in the gallery. For the surveillance system, the concept of redundancy works as follows. At any given surveillance point, the redundancy level is one less than the minimum number of sensors that when failed would result in surveillance failure at the point, i.e. result in the misclassification cost associated with this point to be larger than the prescribed threshold. Redundancy levels at different surveillance points can then be computed individually. For the whole system, all the individual redundancy levels need to be aggregated to represent the whole system. It is not uncommon that the minimum redundancy level is chosen as the (worst-case) fault tolerance representation of the whole surveillance system.

This redundancy measure has a couple of limitations. It is a deterministic measure, not taking into account the uncertain nature of detection and surveillance. This deterministic measure may work well in the art gallery problem, where once a clear line of sight is established, the detection probability is 100%, and false alarms rarely happen. But practical surveillance systems are more complicated than the AGP's. Moreover, a redundancy measure is also an integer number, often spanning a very narrow range, say, one to three (especially true, if the system-wide redundancy is the minimum redundancy of all points). For this reason, the redundancy measure works poorly at discriminating different systems and their robustness, defying the original motivation of devising such a measure.

III. Measures of Fault Tolerance Capability

A. Redundancy Measure

Consider a subset of sensors $X \subseteq I$ that are working according to their designed detection and false alarm probabilities; or equivalently, $I^{(X^-)}$ is the set of the failing sensors whose presence can be excluded from the system. Denote by L_j the maximum allowed level of misclassification cost at surveillance point j, meaning that, when the sensor system produces a misclassification cost smaller than L_j for all $j \in J$, the system is considered working, and otherwise failed.

Denote by $C_j(X)$ the misclassification cost at point j, under surveillance of the sensors in X. Similar to the definition of $C_j(\{i\})$, we can have

$$C_j(X) = c_j^m \cdot [1 - d_j(X)] \cdot P(event) + c_j^f \cdot f_j(X) \cdot P(no\,event), \tag{1}$$

where $d_j(X)$, and $f_j(X)$ are the detection, and false alarm probabilities associated with point j, respectively, but modified to fit the multi-sensor circumstance. Determining $d_j(X)$ and $f_j(X)$ needs to take as inputs the $d_j(\{i\})$ and $f_j(\{i\})$ of individual sensors. But it also is necessary to consider the specific sensor fusion algorithm involved. Consider, for instance, multiple sensors monitoring the same location, and reporting individually "event" or "no-event" on their own. The final fused decision could be to report "event" whenever any one of the sensors reports "event," customarily known as the "1-out-of-n" rule, or report "event" only when all sensors report "event," customarily known as the "n-out-of-n" rule. The detection and false alarm probabilities of combining the sensors in X are going to be different under different decision fusion rules, even if all other conditions are held the same.

Under the setting described above, in particular, given a sensor fusion algorithm, we first present the redundancy measure.

Definition 1: The redundancy level (RL) at surveillance point $j \in J$ is

$$RL(j) = \min\left\{\left|I^{(X^-)}\right| - 1: \quad C_j(X) - L_j > 0, \quad X \subseteq I\right\}. \tag{2}$$

Thus, the redundancy level at a surveillance point is one less than the minimum number of sensors that when failed would result in the misclassification cost associated with this point to be larger than the prescribed threshold.

If aggregating the redundancy levels for individual surveillance points by considering the worst-case scenario, the system-wide redundancy measure is defined as follows.

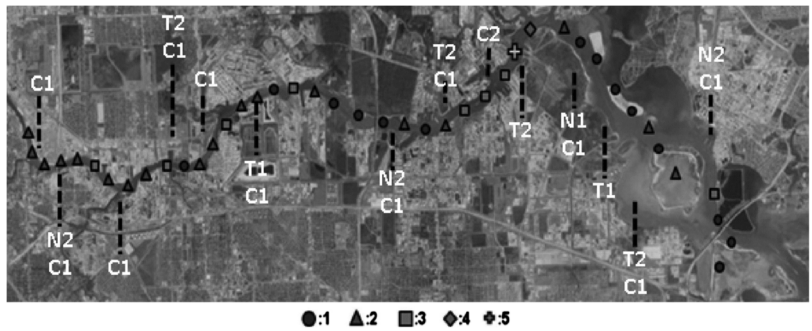

Fig. 1. Redundancy level of a surveillance sensor network at the Houston ship channel. The numbers underneath the figure are the degree of redundancy. T1, C1, etc. represent the sensor types.

Definition 2: The redundancy level (RL) of a surveillance system is

$$RL = \min\left\{ \left| I^{(X^-)} \right| - 1 : \quad \max_{j \in J} (C_j(X) - L_j) > 0, \quad X \subseteq I \right\}. \tag{3}$$

The system-wide redundancy measure is the same as using the smallest redundancy levels among all the surveillance points, namely $RL = \min_j RL(j)$. But note that the failed sensors in $I^{(X^-)}$, defined at the system level, can come from different surveillance points.

We apply this redundancy concept to a surveillance sensor system used in [1], using the sensor fusion rule described in [2] that is designed to minimize the misclassification cost. Fig. 1 illustrates, using markers of various shape combinations, the redundancy levels associated with each surveillance point during daytime; the dashed vertical line marks along the ship channel indicate the locations of the sensor towers. Considering all points together, the smallest individual redundancy level is 1, which becomes the system's redundancy. The redundancy levels characterize individual points with reasonable degree of discrimination. But the system level redundancy measure has much less discriminatory power. It is easy to imagine that another surveillance system, with very different sensors and layouts, may produce the same system-wide redundancy. It then becomes impractical to differentiate different system designs using the redundancy level; it is too simple a measure for a complex system like this.

B. Probabilistic Fault Tolerance Measures

In light of the shortcomings of the redundancy measure, especially its inability to discriminate the networks of sensors at the system level, we hereby propose a probabilistic FTC measure. We define this FTC measure as a conditional probability, characterizing the likelihood that a sensor system is still working in the presence of sensor failures.

Definition 3: The fault tolerance capability of a surveillance system is

$$\begin{aligned} FTC &= Prob(\text{Sensor system working} | \text{at least one sensor failed}) \\ &= Prob\left(C_j(X) - L_j \le 0 \; \forall j \in J | X \subset I\right). \end{aligned}$$

Here, the condition $X \subset I$ implies that $I^{(X^-)} \neq \emptyset$, namely that at least one sensor has failed. Meanwhile, there is no need to consider $I^{(X^-)} = I$, because when all sensors have failed, the probability that the system is still working is zero. Assuming that a sensor's failure is statistically independent from that of the others, Corollary 1 provides a simplified expression for computing this FTC measure.

Corollary 1:

$$FTC = \frac{\sum_{X \subset I, X \neq \emptyset} \left[g(X) \cdot \prod_{i \in X} P(F_i = 0) \cdot \prod_{i \in I^{(X^-)}} P(F_i = 1) \right]}{1 - \prod_{i \in I} P(F_i = 0)}, \tag{4}$$

where

$$g(X) = \begin{cases} 1 & \text{if } C_j(X) \le L_j \text{ for all } j \in J \\ 0 & \text{otherwise.} \end{cases}$$

$$F_i = \begin{cases} 1 & \text{if sensor } i \text{ fails} \\ 0 & \text{otherwise.} \end{cases}$$

In the above expression, $P(F_i = 0)$ is a sensor's (prior) reliability, or equivalently, $P(F_i = 1)$ is the sensor's failure probability. Thus, $\prod_{i \in I} P(F_i = 0)$ is the probability that all sensors are working. $g(X)$ is a logic function checking whether the system's working condition, $C_j(X) \le L_j, \forall j \in J$, is satisfied. Same as in the redundancy measure, this condition needs to be verified for a specific sensor fusion rule.

Last, we define the reliability of a surveillance sensor system. The reliability measure includes the case where all sensors are working.

Definition 4: The reliability of a surveillance system is

$$\begin{aligned} Reliability &= Prob(\text{Sensor system working}) \\ &= Prob\left(C_j(X) - L_j \le 0, \forall j \in J, X \subseteq I\right). \end{aligned}$$

It can be computed as follows.

Corollary 2:

$$Reliability = \sum_{X \subseteq I} \left[g(X) \cdot \prod_{i \in X} P(F_i = 0) \cdot \prod_{i \in I^{(X^-)}} P(F_i = 1) \right] \tag{5}$$

The reliability expression bears resemblance with that of the FTC, except that the reliability is an unconditional probability, giving too much weight to the situation where all sensors are

TABLE I
PROBABILITIES OF DIFFERENT SENSOR FAILURE EVENTS

		X			
		$\{\emptyset\}$	$\{A\}$	$\{B\}$	$\{A,B\}$
Scenario 1	$\prod_{i\in X}P(F_i=0)\cdot\prod_{i\in I^{(X^-)}}P(F_i=1)$	0.01	0.04	0.19	0.76
	$g(X)$	0	1	0	1
Scenario 2	$\prod_{i\in X}P(F_i=0)\cdot\prod_{i\in I^{(X^-)}}P(F_i=1)$	0.01	0.19	0.04	0.76
	$g(X)$	0	1	0	1

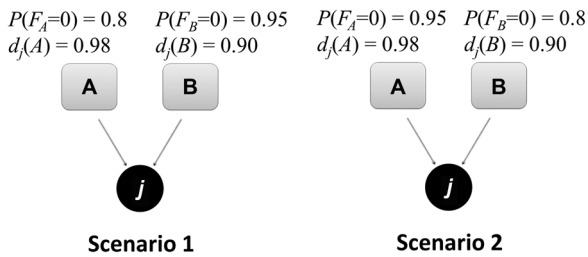

Fig. 2. Example of two sensors, A and B, observing a single surveillance point j; a square represents a sensor, and the circle represents the surveillance point.

working, whereas FTC is a conditional probability, emphasizing the system working probability only after sensor failures take place. While the reliability may serve as the ultimate benchmark of the likelihood that a surveillance system is working, combining the effects from sensors and the system, FTC is a better measure to quantify the robustness of the system designs. We would like to illustrate the difference between the three measures presented above, using a simple two-sensor, single-surveillance-point example, presented in Fig. 2.

In this example, let us suppose that the cost of misdetections (false negatives) significantly outweigh that of false alarms (false positives), and the prior probability of an event is the same as the condition of no event, so that checking the misclassification cost at the surveillance point can be replaced with comparing the detection probability of a sensor or a sensor combination with the desired detection probability. Suppose the cost coefficient $c_j^m = 1$, and set $L_j = 0.03$, which is equivalent to setting the detection probability threshold at 0.97. So $g(X)$ returns 1 if, after removing the sensors in $I^{(X^-)}$, the remaining sensor(s) can still provide a detection probability greater than 0.97, and $g(X)$ returns 0 otherwise.

The sensors are then characterized by two attributes: the sensor's own reliability $P(F_i = 0)$, and its detection probability $d_j(\cdot)$. The two scenarios presented in Fig. 2 represent different combinations of sensor reliability and detection probability. Table I presents the probabilities under different sensor failure events as well as the outcome of $g(X)$, indicating whether the system is still working, under the corresponding failure event. These quantities are the ingredients for computing the FTC, and reliability measures using (4) and (5), respectively. The redundancy level is also computed, using (2). The results are included in Table II.

The interpretation of the probabilistic measures can be understood as follows. Take Scenario 1 as an example. Suppose there are 100 randomly selected cases of the sensor network in Scenario 1. On average, there will be 76 cases of no sensor failure, 19 cases of sensor A failed; four cases of sensor B failed, and one case of both failed. Among the 19 cases of sensor A failed, the system is still working in none of these cases because $d_j(B)$ is too low. For the four cases where B failed, the system is still working in all of them because A provides the required level of detection. For the single case of both sensors failed, the system also failed. So, for the 24 sensor failing cases, the system is still working in four of them, translating to a 17% or 0.17 FTC (rounded to the second decimal). The reliability of the system also considers the 76 cases when both sensors are working, under which the system also works, translating to a total of 80 system working cases out of 100 cases (i.e., $Reliability = 0.80$).

TABLE II
DIFFERENT MEASURES OF FAULT TOLERANCE IN THE TWO-SENSOR EXAMPLES

	Scenario 1	Scenario 2
RL	0	0
FTC	0.17	0.79
Reliability	0.80	0.95

Scenario 2 is where we swap the two sensors' reliabilities. It turns out the FTC of the system drastically increases from 0.17 to 0.79 by this simple action. This result appears to make sense because, in the previous scenario, the more capable sensor is less reliable, while here, the more capable sensor is more reliable; pairing capability with reliability apparently makes a big difference.

The reliability of the system also increases to 0.95, due to the new design of the system, because all the prior reliabilities of the sensors are unchanged. The increase demonstrated by the system reliability measure is, however, much less pronounced than the FTC measure. The decrease in the system failure probabilities, from 0.20 for Scenario 1 to 0.05 for Scenario 2, appears more pronounced, having reduced to one-fourth. However, this change is much less pronounced compared to the FTC measure, especially when the absolute differences are compared. This change in the system reliability or failure probability is less pronounced because the inclusion of the 76% cases of no sensor failures dilutes the effect coming from the system change. We would like to note this stronger indicator as one merit of using the FTC.

It turns out that, in both Scenarios 1 and 2, the system's reliability is the same as the reliability of the more capable sensor, namely that whenever the capable sensor works, the system works. Although this makes intuitive sense, it does cast a little

doubt on how much additional value is added by this system reliability as a new measure.

IV. Computing the Fault Tolerance Capability

As noted in Section III, there could be many sensor fusion rules (for example, a k-out-of-n rule, where k is between 1 and n). The fault tolerance measure defined and discussed in Section III characterizes a surveillance system's fault tolerance capability under a specific decision rule. The study of decision rules falls in the large body of literature on sensor fusion as we reviewed in Section II. That topic is not the focus of our paper. Here, we assume that the sensor fusion rule is given, and more specifically, choose to use the rule presented in [2], for two reasons: (a) the rule in [2] is designed to *minimize* the misclassification cost; and (b) when compared to a good number of other decision rules, including the k-out-of-n rule with an optimized k, the rule in [2] appears to be superior. This superiority is particularly observed when the sensors to be fused are heterogeneous, which is something that frequently happens in surveillance settings.

To compute the probabilistic fault tolerance and reliability measures using (4) and (5), the part of checking the logic function $g(X)$ presents a technical challenge. Because a surveillance point can be observed by multiple sensors, and each sensor can observe multiple surveillance points, the sensor network looks like an intertwining bipartite graph that is not so easy to be dissected to isolated pieces. Presumably, one would need to sort through all possible $X \subset I$ to decide, for each X, whether $g(X) = 1$ or $g(X) = 0$. When the sensors in I are plenty, the total number of subsets one needs to exhaust can be a large quantity; for instance, when $n = 41$ as in some instances of the surveillance system in [2], the number of subsets is about 2 trillion.

Computing the redundancy measure is considerably simpler because $RL = min_j RL(j)$. One can compute the $RL(j)$ individually, and then select the smallest value as RL. Computing the individual $RL(j)$ is affordable because the number of sensors per point is generally a single-digit number. A similar relationship, unfortunately, does not exist for the FTC and reliability measures. One can construct simple examples to show that the system-wide FTC could be larger or smaller than the smallest of the $FTC(j)$.

We in this section present a structured search to avoid the complete enumerations for computing the FTC and reliability measures. Readers going through the simple two-sensor examples in Section III probably have already considered that not all the sensors are equivalent; the capable sensors appear, in the mission of ensuring a working surveillance system, to matter much more than the inferior ones. Our basic idea for reducing the number of sensor subsets one needs to evaluate for checking $g(X)$ is intuitive: devise a quantitative index ranking sensors' or sensor combinations' capabilities; and only check $g(X)$ under the failure of much smaller subsets of sensors or sensor combinations whose capability is low, and to see whether their removal will cause the whole system to fail. Once a (failed) subset of sensors is found to cause the system to fail, then any subset formed by replacing one or more sensor with a more capable one, or adding sensors to the set, will also cause system failure.

In the two-sensor example, each sensor's capability is simply characterized by its detection probability. The index to use for an actual system is necessarily more complicated in two aspects: (a) both detection and false alarm probabilities need to be considered; and (b) the index can be applied to not only a single sensor but also a multi-sensor combination, and represents the combined capability of the sensor combination.

The second aspect suggests that the sensor ranking index we are seeking depends on the specific sensor fusion rule used. In fact, when working out their sensor fusion rule, [2] have already devised an index testifying the sensor's capability. Next we will explain the index, and show that it fits our use in computing the FTC and reliability measures. In explaining the sensor ranking index, it becomes obvious that we will have to repeat certain notations and expressions from [2] to make this paper self-contained.

A. Sensor Ranking Index, Structural Properties, and Algorithm Development

Denote by U_j the actual occurrence of events at surveillance point j, and by Y_{ij} the decision of sensor i (or the decision from the sensor algorithm-operator combo) observing point j. Both the occurrence and the decision are binary, with 1 representing the occurrence of an event, and 0 representing no event occurred. As such, we can express $1 - d_j(\{i\}) = P(Y_{ij} = 0|U_j = 1)$, and $f_j(\{i\}) = P(Y_{ij} = 1|U_j = 0)$.

Denote by $\mathbf{y}_j = (y_{1j}, \ldots, y_{n_j j})$ the vector of a realization of $\mathbf{Y}_j = (Y_{ij}, \ldots, Y_{n_j j})$. The fusion algorithm takes all the values in $\mathbf{y}_j$, and returns a 0 or 1. Denote this sensor fusion outcome by $x(\mathbf{y}_j)$. Using this set of notations, the misclassification cost can be expressed as

$$C_j(I) = \sum_{\{\mathbf{y}_j : x(\mathbf{y}_j)=1\}} \left[c_j^f \cdot P\left(x(\mathbf{y}_j)=1|U_j=0\right) \cdot P(U_j=0)\right] + \sum_{\{\mathbf{y}_j : x(\mathbf{y}_j)=0\}} \left[c_j^m \cdot P\left(x(\mathbf{y}_j)=0|U_j=1\right) \cdot P(U_j=1)\right], \quad (6)$$

where $\{\mathbf{y}_j : x(\mathbf{y}_j) = 1\}$ is the set of $\mathbf{y}_j$ for which the fused outcome is 1, and $\{\mathbf{y}_j : x(\mathbf{y}_j) = 0\}$ is the set of $\mathbf{y}_j$ for which the fused outcome is 0. The argument in $C_j(\cdot)$ is I, meaning that this sensor fusion considers the decisions from all sensors involved. The optimal fusion rule in [2], denoted by $x^*(\mathbf{y}_j)$, is obtained by minimizing $C_j(I)$. This optimal fusion rule can be obtained through the following condition. $x^*(\mathbf{y}_j) = 1$ if

$$\prod_{i \in S(\mathbf{y}_j)} \delta_{ij} \leq \beta_j, \quad (7)$$

such that

$$S(\mathbf{y}_j) = \{i \in I : y_{ij} \neq 1\},$$
$$\delta_{ij} = \frac{P(Y_{ij}=1|U_j=1)}{P(Y_{ij}=1|U_j=0)} \cdot \frac{P(Y_{ij}=0|U_j=0)}{P(Y_{ij}=0|U_j=1)}, \text{and}$$
$$\beta_j = \frac{c_j^m P(U_j=1)}{c_j^f P(U_j=0)} \prod_{i \in I} \frac{P(Y_{ij}=1|U_j=1)}{P(Y_{ij}=1|U_j=0)}.$$

Procedural details are omitted because they can be found in [2]. When presenting the above condition in their original expression, [2] had used an auxiliary variable y_{ij}^*, which equals 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 0 otherwise. We believe that a sensor with a false alarm or misdetection probability greater than 0.5 is not fit for use for surveillance purposes. As a matter of fact, we seldom encountered surveillance systems that used such inferior sensors. So here we assume that a sensor's false alarm and misdetection probabilities are both smaller than or equal to 0.5. Therefore, we have that $P(Y_{ij}=1|U_j=1) \geq P(Y_{ij}=0|U_j=1)$, and $P(Y_{ij}=0|U_j=0) \geq P(Y_{ij}=1|U_j=0)$. Consequently, y^*_{ij} is always 1. With this result, we attain the above simplified result from the original expression in [2]. [2] interpreted the result in (7) as "... β_j *can be considered as a performance threshold determined by the sensor system and the surveillance task. On the left hand side*, δ_{ij} *represents the capability of sensor* i (i.e., *false alarm rate and detection power*)."

We add two additional remarks regarding δ_{ij}. (a) First, the larger δ_{ij} is, the better. Considering only sensors whose false alarm and misdetection probabilities smaller than 0.5, we have $\delta_{ij} \geq 1$. (b) Because of the product in front of δ_{ij} in (7), the above condition applies to a multi-sensor combination. The δ_{ij} will serve as our sensor ranking index that helps the computing of FTC and reliability measures.

Now, suppose that sensor $k \in I$ failed. So $I^{(k^-)}$ is the set of sensors after excluding sensor k. Then represent the sensor outputs associated with point j after the exclusion of sensor k by $\tilde{\mathbf{y}}_j = (\tilde{y}_{1j}, \tilde{y}_{2j}, \ldots, \tilde{y}_{(k-1)j}, \tilde{y}_{(k+1)j}, \ldots, \tilde{y}_{n_j j})$. Moreover, create two artificial decision vectors $\mathbf{y}^0_j$ and $\mathbf{y}^1_j$, such that $y^0_{ij} = y^1_{ij} = \tilde{y}_{ij}\ \forall i \neq k$, and $y^0_{kj} = 0$, $y^1_{kj} = 1$. Lemma 3 provides the optimal fusion $x^*(\tilde{\mathbf{y}}_j)$ when $x^*(\mathbf{y}^0_j) = x^*(\mathbf{y}^1_j)$. See the Appendix for the proof.

Lemma 3: If $x^*(\mathbf{y}^0_j) = x^*(\mathbf{y}^1_j)$, then $x^*(\tilde{\mathbf{y}}_j) = x^*(\mathbf{y}^0_j) = x^*(\mathbf{y}^1_j)$, and $C_j(I^{(k^-)}) = C_j(I)$.

Lemma 3 can be understood as follows. If the fused decision does not change compared to the decision that could have been made by including sensor k, then it means that the fused decision using the sensor from $I^{(k^-)}$ can be attained by arbitrarily inserting any output for sensor k. Consequently, under this circumstance, using the decision vector $\tilde{\mathbf{y}}_j$ does not alter the misclassification cost for point j.

Next, we consider the relation between $C_j(I^{(k^-)})$ and $C_j(I)$ when $x^*(\mathbf{y}^0_j) \neq x^*(\mathbf{y}^1_j)$. From the definition of $S(\mathbf{y}_j)$, it is clear that $k \notin S(\mathbf{y}^1_j)$ because $y^1_{kj} = 1$, and $k \in S(\mathbf{y}^0_j)$ because $y^0_{kj} = 0$. Also, because all $\delta_{i,j} \geq 1$, we have

$$\prod_{i \in S(\mathbf{y}^1_j)} \delta_{ij} \leq \delta_{kj} \cdot \prod_{i \in S(\mathbf{y}^1_j)} \delta_{ij} = \prod_{i \in S(\mathbf{y}^0_j)} \delta_{ij}.$$

Using the sensor fusion condition in (7), the above inequality suggests that, if $x^*(\mathbf{y}^0_j) = 1$, then $x^*(\mathbf{y}^1_j) = 1$, or in other words, for all $x^*(\mathbf{y}^0_j) \neq x^*(\mathbf{y}^1_j)$, $x^*(\mathbf{y}^0_j) = 0$, and $x^*(\mathbf{y}^1_j) = 1$.

Lemma 4: When $x^*(\mathbf{y}^0_j) \neq x^*(\mathbf{y}^1_j)$, then $x^*(\mathbf{y}^0_j) = 0$, and $x^*(\mathbf{y}^1_j) = 1$. Also $C_j(I^{(k^-)}) \geq C_j(I)$.

Lemma 4 describes the situation where the output of sensor k matters. Under this circumstance, it is quite understandable that a failing sensor k could increase the misclassification cost $C_j(\cdot)$.

As we mentioned before, not all sensors are equally important. Some sensors matter in the final fused decision (Lemma 4), thereby affecting the misclassification cost, while others do not matter at all (Lemma 3), the exclusion of which leaves the misclassification cost unchanged. Apparently, a sensor of large δ_{kj} (capable sensor) is more likely to matter, while a small δ_{kj} identifies a less capable sensor whose failure is more likely to be tolerated.

Combining Lemma 3 and Lemma 4, we conclude that $C_j(I^{(k^-)}) \geq C_j(I)$; that is, the misclassification cost is non-decreasing in the number of sensor failures. Therefore, if the failure of a subset of sensors results in a misclassification cost larger than the prescribed L_j for some surveillance point j, then the failure of any subset of sensors that contains this subset will surely also lead to a misclassification cost larger than L_j.

Based on the above understanding, we outline the procedure for computing the FTC and reliability measures as follows. The procedure is to find all the sets of sensors and sensor combinations whose removal (i.e. failure) still retains a $g(X) = 1$, and then compute FTC and $Reliability$ using the sets found.

Algorithm: Computing FTC, and $Reliability$

1) Re-index the sensors such that $\max_{j \in J} \delta_{(i-1)j} \leq \max_{j \in J} \delta_{(i)j}$ for $i = 2, 3, \ldots, n$.
2) Let $\mathbf{Z}^{(t)}$ be the set of t-sensor combinations, and set $\mathbf{Z}^{(1)} = \emptyset$.
3) Let $t = 1$; this step investigates a single sensor failure.
   - For $i = 1, \ldots, n$, run the following.
     ◇ Let $I^{(X^-)} = \{i\}$.
     ◇ If $C_j(X) \leq L_j, \forall j \in J$, then $g(X) = 1$, and $\mathbf{Z}^{(1)} = \mathbf{Z}^{(1)} \cup I^{(X^-)}$.
     ◇ Else break the `for-loop`.
   - If $\mathbf{Z}^{(1)} = \emptyset$, then $FTC = 0$, $Reliability = 0$, and Stop; otherwise, continue.
4) Let $t = t + 1$, and set $\mathbf{Z}^{(t)} = \emptyset$. The next step adds one sensor at a time to an existing $(t-1)$-sensor combination (from $\mathbf{Z}^{(t-1)}$) to make a new t-sensor combination, and then check whether the t-sensor failure can be tolerated by the system.
5) For $\ell = 1, \ldots, |\mathbf{Z}^{(t-1)}|$, select $Z^{(t-1)}_\ell \in \mathbf{Z}^{(t-1)}$; and for $i \in \{1, \ldots, n\} \setminus Z^{(t-1)}_\ell$, run the following.
   - Let $I^{(X^-)} = \{i, Z^{(t-1)}_\ell\}$; this makes a t-sensor combination.
   - If $C_j(X) \leq L_j, \forall j \in J$, then $g(X) = 1$, and $\mathbf{Z}^{(t)} = \mathbf{Z}^{(t)} \cup I^{(X^-)}$; otherwise, break the inner `for-loop` indexed by i.
6) If $\mathbf{Z}^{(t)} = \emptyset$, compute FTC and $Reliability$ using the sensor sets included in $\{\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(t-1)}\}$, and Stop; otherwise, go to Step 4.

The above algorithm ranks the sensors from the inferior to superior according to their δ value. Then in Step 3 and Step 5, when sorting through all individual sensors (the `for-loop` indexed by i), we will check whether, when an inferior sensor is removed, the system will stop working. If yes, then there is no need to check any other superior sensors, because we know the removal of a superior sensor will definitely cause the system to fail.

B. Applicability and Efficiency of the Algorithm

The structural properties and the subsequent algorithm presented in the previous subsection were derived based on the choice of a specific sensor fusion rule and the associated ranking index. So a natural question is what happens if a different sensor fusion rule is used, or given a different sensor fusion rule, can the sensor ranking index presented above still be used?

We believe that our choice of the decision fusion rule is representative of a class of (popular, good performing) decision fusion rules, inherently possessing a sensor ranking index with the following two properties:

1) **Ordering property**: If a certain sensor subset causes system failure, then replacing any sensor in this combination by a superior sensor will cause system failure.
2) **Hierarchy property**: If a certain sensor combination causes system failure, then any set containing this set will also cause system failure.

These properties make intuitive sense. Given a different decision fusion rule, one should check whether the presented sensor ranking index can work with it, and then whether the two properties are satisfied.

The hierarchy property described above can be compared to the concept of coherent systems in systems reliability. A system is called coherent if, given the system has failed after the failure of a set of components, the system remains failed in the event of additional component failures. Thus, a surveillance system equipped with a sensor fusion algorithm satisfying the hierarchy property is a coherent system. The reliability of such a system can be computed by finding the minimal cuts, i.e., the minimal sets of components (or sensors, for a surveillance system) whose failure results in system failure. The hierarchy property can be understood as saying that any set containing a cut set is still a cut set; this is a well-established understanding from the system reliability literature. And the algorithm in Section IV-A essentially accomplishes the same task, but more efficiently by also using the ordering property of the sensor fusion algorithm, stated above.

Not all sensor fusion rules use a sensor ranking index. The commonly used version of the k-out-of-n rule does not differentiate individual sensors. More importantly, for a fixed k, the k-out-of-n rule does not guarantee to produce the smallest misclassification cost. It is possible that k-out-of-$(n-1)$ (after one sensor failure) produces a smaller misclassification cost than k-out-of-n. When this happens, the non-decreasing property implied by Lemmas 3 and 4 breaks down, and the hierarchy property expressed above will not hold.

A revised version of the k-out-of-n rule is the optimal-k rule, which is to, instead of using a fixed k, exhaust all choices of k between 1 and n, and select k^* that minimizes the misclassification cost. Following this rule eliminates the cause of violation of the non-decreasing property mentioned above. Then, this optimal-k rule can be paired with the presented sensor ranking index for the purpose of selecting which subset of k^* sensors to make the final fusion. As a result, those k^* sensors matter, while the other $n-k^*$ sensors do not. So the structured search algorithm works.

TABLE III
TEST INSTANCES FROM [1]

N	$\|J\|$	n	Min $\|J_i\|$	Avg. $\|J_i\|$	Max $\|J_i\|$	Min n_j	Avg. n_j	Max n_j
1	42	23	4	6.3	10	2	4	6
2	42	26	4	5.7	7	2	4	6
3	84	38	4	6.1	12	2	3	5
4	84	40	4	5.9	17	2	3	5
5	42	25	4	6.3	10	2	4	6
6	42	23	4	6.5	10	2	4	6
7	84	41	4	6.4	17	2	4	6
8	84	41	4	6.5	17	2	4	5
9	42	22	4	6.2	10	2	4	6
10	42	26	3	5.5	8	2	4	7
11	84	37	3	5.9	17	2	3	5
12	84	40	3	5.6	17	2	3	4
13	42	22	4	6.5	11	2	4	6
14	42	24	4	6.3	10	2	4	6
15	84	40	4	6.2	17	2	3	6
16	84	39	4	6.2	17	2	3	5

The remaining question regarding algorithm efficiency is whether the structured search always sorts through only a small number of sensor combinations. Without any other practical constraints, the answer is probably no. In theory, people could construct examples where the search needs to exhaust a large number of sensor combinations. On the other hand, however, we believe the structured search will be very efficient for practical surveillance systems.

What we noticed in typical surveillance systems is that (a) albeit different, sensors are generally capable, so it is unlikely that people can remove a bunch of sensors while the system's misclassification cost remains unchanged; and (b) the degree of redundancy is not high. These two features should not come as a surprise. Given the critical role that surveillance systems play, people would expect their designer to be thoughtful when choosing the types and number of sensors. And these sensors are expensive, especially considering the cost not only of the sensors but also of installation and operation. These two features of surveillance systems ensure that the number of sensor sets in $\{\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(t-1)}\}$ is small; so one would expect to stop at a relatively small i as well as a small t. Empirically, as shown in the subsequent section, the above algorithm often returns around 0.5% of the total sensor combinations for computing FTC and $Reliability$.

V. CASE STUDY

We show in this section how the fault tolerance measures are applied to a surveillance sensor system on the Houston ship channel. [1] generated optimal surveillance system layouts for 16 different realistic instances of the surveillance system. We use these 16 instances in our study. Table III summarizes the information of the 16 instances, where N denotes the instance number.

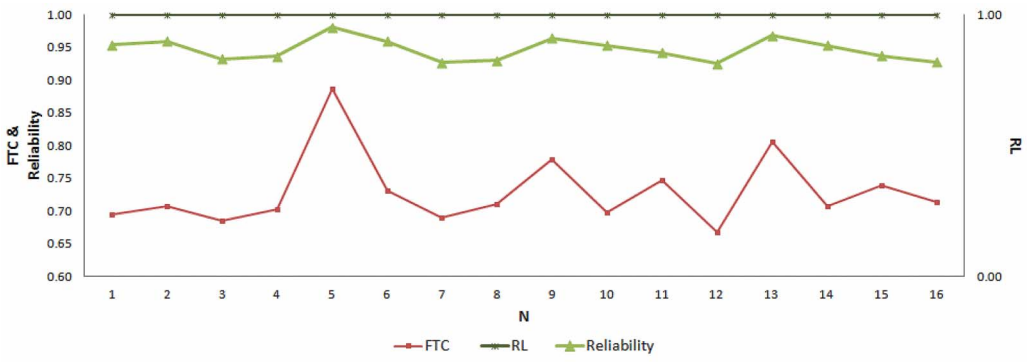

Fig. 3. RL, $Reliability$, and FTC measures for each of the 16 surveillance system instances, with the horizontal axis as the system instance.

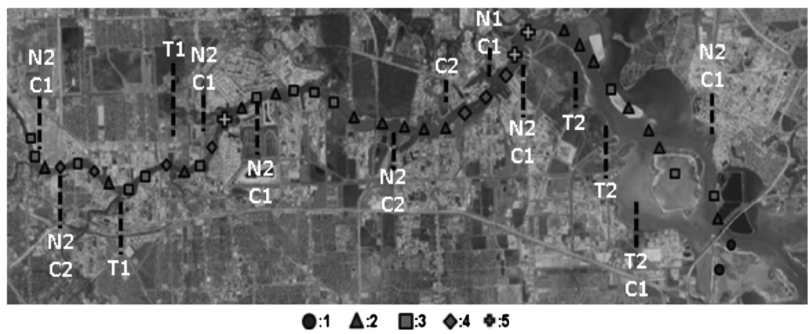

Fig. 4. Redundancy levels of the surveillance points in the surveillance system instance # 5.

As in [2], we use the misdetection probabilities (MDP) and sensor failure probabilities given in [1]. The false alarm probabilities (FAP) and the cost coefficients were not specified in [1]. [2] compared their decision fusion rule for various choices of the false alarm probability, both fixed and varying with the misdetection probability, as well as different choices of the cost coefficients. As far as the decision fusion rule is concerned, the conclusions were similar. The fault tolerance measures can certainly be computed under any cost or probability settings, using the same procedure. But for simplicity of presentation, we choose to present the results under FAP=MDP and $c_j^m/c_j^f = 1$.

Regarding the system working requirement, we set $L_j = 0.03$ (or $1 - L_j = 0.97$), because [1] generates the surveillance system layouts based on the assumption that the surveillance system should detect an intrusion at each surveillance point with an average probability of 0.97.

The procedure presented in Section IV helped remarkably in reducing the time for computing FTC and $Reliability$. For the 16 cases, on average, 99.47% of the sensor combination subsets were skipped for evaluation, or equivalently, only 0.53% of the total sensor combinations, on average, are needed to compute FTC and $Reliability$. Among the 16 surveillance system instances, the longest time required for computing the FTC, using the proposed procedure, is 965 seconds (or 16.1 minutes). For that case, only 0.02% of the sensor combinations were evaluated. Had all the sensor combinations been evaluated, and assuming that the time scales linearly, the computation time would extrapolate to 55 days.

Comparison of the FTC, $Reliability$, and RL measures are presented in Fig. 3. One can see that $RL = 1$ for all 16 surveillance instances, confirming our previous argument that the redundancy level is not a discriminating metric that can differentiate well between system designs of different layouts and sensor combinations. The FTC and $Reliability$ follow a similar trend, but the FTC exhibits a much more pronounced pattern of change between different instances than $Reliability$ because FTC focuses on the situations when some sensors fail, while $Reliability$ gives more weight to the situation where all sensors are working. There are also a few pairs of instances in Fig. 3 where the FTC and $Reliability$ trends do not match. For example, consider instances # 10 and # 11. Clearly, the FTC for instance # 11 is greater, but the $Reliability$ of # 10 is greater. This highlights the fact that a more reliable system is not necessarily more robust, i.e., it may not continue to be the most reliable system after a sensor has failed.

The surveillance system displayed in Fig. 1 corresponds to instance # 1, which according to Fig. 3 is among the systems of relatively low FTC values. Instance # 5, on the other hand, has the highest FTC among the sixteen instances (0.89 in # 5 versus 0.69 in # 1). So we display in Fig. 4 the surveillance system layout as well as the redundancy level associated with each surveillance point in instance # 5. The sensor locations in this system are the same as those in instance # 1 because, once the sensor towers were built, it is costly to relocate them. But people can change sensor installations on each tower. Instance # 5 does have different sensor installations at a number of sensor towers. Instance # 5 also has two more sensors, for which people selected more capable sensors with longer range and lower failure rates. The different design made a remarkable difference. Looking at the redundancy level at individual

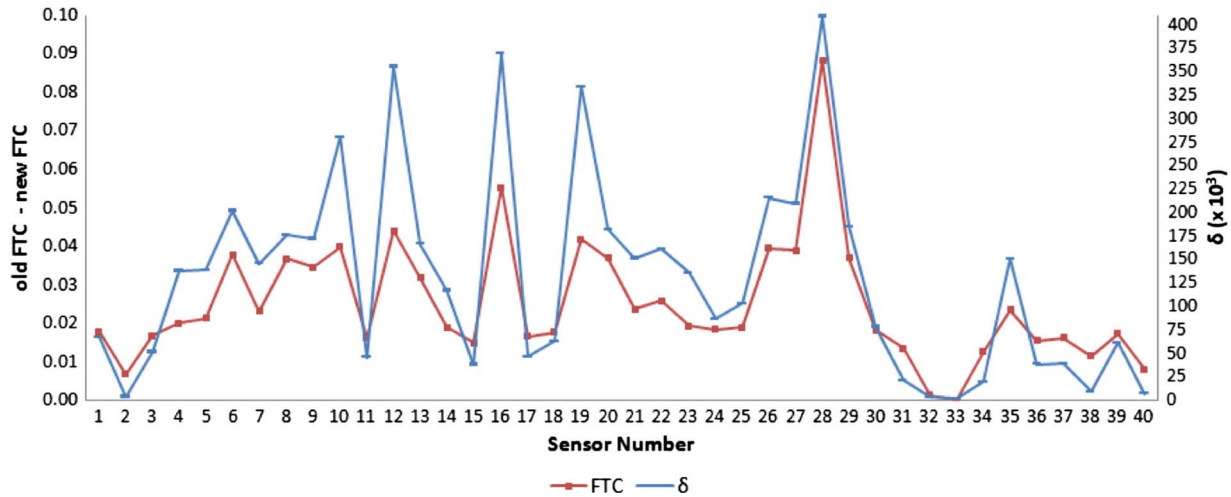

Fig. 5. Impact of individual sensors on the system's fault tolerance capability.

surveillance points, instance # 5 has fewer points whose $RL(j)$ are 1, and more points whose $RL(j)$ are 4 or 5. This in the end translates to a more robust system overall.

We also analyze the impact of individual sensors on the system's fault tolerance capability. We choose instance # 12, and observe the FTC's change upon removing one sensor at a time. With 40 sensors in the original system, $FTC = 0.67$. Fig. 5 shows the decrease in FTC after sensor reduction, which is "old FTC—new FTC." Together with this change in FTC is the δ value associated with each sensor. We observe that some sensors hardly reduce FTC at all, while some other sensors can reduce FTC by as much as 0.10 (that is, 15% of the original FTC). This result echoes our arguments surrounding the two lemmas, i.e., some sensors matter, and some others do not. The figure also shows a strong positive correlation between the δ index with the magnitude of reduction in FTC, as we indicated before.

VI. Concluding Remarks

We believe that the main contribution of this paper is two-fold: (a) we propose a new fault tolerance capability measure for surveillance sensor systems, and present an efficient algorithm to compute it for practical applications; and (b) although the concept of reliability is not new, we present a specific definition of $Reliability$ for surveillance sensor systems, and address its computation issue together with that for FTC. We advocate the use of the FTC measure as the chief metric quantifying the robustness of surveillance systems because of its ability to emphasize the system's response in the presence of sensor failures.

What we did not do in this paper is optimize a surveillance system for the highest possible FTC. That objective is definitely worthy, but doing so warrants a separate effort of research. When attempting to optimize the robustness of a surveillance system, we believe the proposed FTC measure, rather than the redundancy or reliability measures, should be used as the performance metric, due to its articulating reflection of the robustness change in systems. Regardless of which measure is used in an optimization, the fast computation algorithm for the measures devised in this paper is helpful.

A byproduct of our research is the δ index, originally introduced in [2], and its use in ranking sensors' capability. A simple rule of thumb in making a system more robust is to make a capable sensor more reliable, or pair a capable sensor with another equally capable sensor.

Appendix

Proof of Lemma 3: According to Theorem 1 in [2], if $x^*(\mathbf{y}_j^0) = x^*(\mathbf{y}_j^1) = 1$, then

$$c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 0\right) P(U_j = 0) \le c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 1\right) P(U_j = 1),$$

and

$$c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 0\right) P(U_j = 0) \le c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 1\right) P(U_j = 1).$$

Also, because

- the sensors in I are statistically independent,
- $P(Y_{kj} = 0|U_j = 0) + P(Y_{kj} = 1|U_j = 0) = 1$, and
- $P(Y_{kj} = 0|U_j = 1) + P(Y_{kj} = 1|U_j = 1) = 1$,

we have

$$\begin{aligned}
&c_j^f P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 0) P(U_j = 0) \\
&= c_j^f P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 0) \\
&\quad \times [P(Y_{kj} = 0|U_j = 0) + P(Y_{kj} = 1|U_j = 0)] P(U_j = 0) \\
&= c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 0\right) P(U_j = 0) \\
&\quad + c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 0\right) P(U_j = 0) \\
&\le c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 1\right) P(U_j = 1) \\
&\quad + c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 1\right) P(U_j = 1) \\
&= c_j^m P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 1) \\
&\quad \times [P(Y_{kj} = 0|U_j = 1) + P(Y_{kj} = 1|U_j = 1)] P(U_j = 1) \\
&= c_j^m P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 1) P(U_j = 1).
\end{aligned}$$

Hence, $x^*(\tilde{\mathbf{y}}_j) = 1$, again from Theorem 1 in [2].

Alternatively, if $x^*(\mathbf{y}_j^0) = x^*(\mathbf{y}_j^1) = 0$, then

$$c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 1\right) P(U_j = 1) \leq c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 0\right) P(U_j = 0)$$

and

$$c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 1\right) P(U_j = 1) \leq c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 0\right) P(U_j = 0).$$

Similar to the case where $x^*(\mathbf{y}_j^0) = x^*(\mathbf{y}_j^1) = 1$, we can show that

$$c_j^m P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 1) P(U_j = 1) \leq c_j^f P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 0) P(U_j = 0),$$

which leads to $x^*(\tilde{\mathbf{y}}_j) = 0$.

To show the misclassification cost does not change, we first present an alternative expression of (6). It is noted in [2] that, for given $\mathbf{y}_j$, the false alarm cost is incurred only if $x(\mathbf{y}_j) = 1$, and the misdetection cost is incurred only if $x(\mathbf{y}_j) = 0$. As such,

$$C_j(I) = \sum_{\mathbf{y}_j \in \{0,1\}^{n_j}} \Big[c_j^f \cdot P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 0) \cdot P(U_j = 0) x(\mathbf{y}_j) + c_j^m \cdot P(\mathbf{Y}_j = \mathbf{y}_j | U_j = 1) \cdot P(U_j = 1)\left(1 - x(\mathbf{y}_j)\right) \Big]. \quad (8)$$

Using the notation of $\mathbf{y}_j^0$ and $\mathbf{y}_j^1$, we can have

$$\begin{aligned} C_j(I) = \sum_{\tilde{\mathbf{Y}}_j \in \{0,1\}^{n_j}} \Big[& c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 0\right) P(U_j = 0) x^*\left(\mathbf{y}_j^0\right) \\ & + c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 1\right) P(U_j = 1)\left(1 - x^*\left(\mathbf{y}_j^0\right)\right) \\ & + c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 0\right) P(U_j = 0) x^*\left(\mathbf{y}_j^1\right) \\ & + c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 1\right) P(U_j = 1)\left(1 - x^*\left(\mathbf{y}_j^1\right)\right) \Big], \end{aligned}$$

while on the other hand,

$$\begin{aligned} & C_j\left(I^{(k^-)}\right) \\ & = \sum_{\tilde{\mathbf{Y}}_j \in \{0,1\}^{n_j}} \Big[c_j^f \cdot P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 0) \cdot P(U_j = 0) x^*(\tilde{\mathbf{Y}}_j) \\ & \qquad + c_j^m \cdot P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 1) \\ & \qquad \cdot P(U_j = 1)\left(1 - x^*(\tilde{\mathbf{Y}}_j)\right) \Big]. \end{aligned} \quad (9)$$

From the previous steps of proving this Lemma, we can also have the following (trying either $x^*(\mathbf{y}_j^0) = x^*(\mathbf{y}_j^1) = x^*(\tilde{\mathbf{y}}_j) = 0$, or $x^*(\mathbf{y}_j^0) = x^*(\mathbf{y}_j^1) = x^*(\tilde{\mathbf{y}}_j) = 1$).

$$\begin{aligned} & c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 0\right) P(U_j = 0) x^*\left(\mathbf{y}_j^0\right) \\ & \quad + c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 1\right) P(U_j = 1)\left(1 - x^*\left(\mathbf{y}_j^0\right)\right) \\ & \quad + c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 0\right) P(U_j = 0) x^*\left(\mathbf{y}_j^1\right) \\ & \quad + c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 1\right) P(U_j = 1)\left(1 - x^*\left(\mathbf{y}_j^1\right)\right) \\ & = c_j^f P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 0) P(U_j = 0) x^*(\tilde{\mathbf{Y}}_j) \\ & \quad + c_j^m P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 1) P(U_j = 1)\left(1 - x^*(\tilde{\mathbf{Y}}_j)\right) \end{aligned}$$

This concludes the proof.

Proof of Lemma 4: In the discussion leading to the lemma, we have already shown that, if $x^*(\mathbf{y}_j^0) \neq x^*(\mathbf{y}_j^1)$, then $x^*(\mathbf{y}_j^0) = 0$ and $x^*(\mathbf{y}_j^1) = 1$. So what remains to be shown is that $C_j(I^{(k^-)}) \geq C_j(I)$.

Given that $x^*(\mathbf{y}_j^0) = 0$, and $x^*(\mathbf{y}_j^1) = 1$, $C_j(I)$ simplifies as

$$C_j(I) = \sum_{\tilde{\mathbf{Y}}_j \in \{0,1\}^{n_j}} \Big[c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 1\right) P(U_j = 1) + c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 0\right) P(U_j = 0) \Big].$$

Next, we want to show that the term inside the bracket is no greater than the corresponding term in (9). To see this, consider

$$\begin{aligned} & c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 1\right) P(U_j = 1) \\ & \quad + c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 0\right) P(U_j = 0) \\ & \leq c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 0\right) P(U_j = 0) x^*(\tilde{\mathbf{Y}}_j) \\ & \quad + c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^0 | U_j = 1\right) P(U_j = 1)\left(1 - x^*(\tilde{\mathbf{Y}}_j)\right) \\ & \quad + c_j^f P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 0\right) P(U_j = 0) x^*(\tilde{\mathbf{Y}}_j) \\ & \quad + c_j^m P\left(\mathbf{Y}_j = \mathbf{y}_j^1 | U_j = 1\right) P(U_j = 1)\left(1 - x^*(\tilde{\mathbf{Y}}_j)\right) \\ & = c_j^f P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 0) P(U_j = 0) x^*(\tilde{\mathbf{Y}}_j) \\ & \quad + c_j^m P(\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j | U_j = 1) P(U_j = 1)\left(1 - x^*(\tilde{\mathbf{Y}}_j)\right). \end{aligned}$$

To see the inequality, simply try either $x^*(\tilde{\mathbf{y}}_j) = 0$, or $x^*(\tilde{\mathbf{y}}_j) = 1$. The equality follows by using the same trick as in the proof of Lemma 3 (recall the statistical independence assumption regarding sensors).

REFERENCES

[1] 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.

[2] E. I. Gokce, A. K. Shrivastava, J. J. Cho, and Y. Ding, "Decision fusion from heterogeneous sensors in surveillance sensor systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 1, pp. 228–233, Feb. 2011.

[3] W. Kuo and M. J. Zuo, *Optimal Reliability Modeling: Principles and Applications*. New York, NY, USA: Wiley, 2002.

[4] Y. Ali and S. Narasimhan, "Sensor network design for maximizing reliability of linear processes," *AIChE J.*, vol. 39, pp. 820–828, 1993.

[5] H. M. F. AboElFotoh, S. S. Iyengar, and K. Chakrabarty, "Computing reliability and message delay for cooperative wireless distributed sensor networks subject to random failures," *IEEE Trans. Re.*, vol. 54, no. 1, pp. 145–155, Mar. 2005.

[6] Q. Yang and Y. Chen, "Sensor system reliability modeling and analysis for fault diagnosis in multistage manufacturing processes," *IIE Trans.*, vol. 41, pp. 819–830, 2009.

[7] R. Viswanathan and P. K. Varshney, "Distributed detection with multiple sensors: Part 1—Fundamentals," *Proc. IEEE*, vol. 85, no. 1, pp. 54–63, Jan. 1997.

[8] R. S. Blum, A. Kassam, and H. V. Poor, "Distributed detection with multiple sensors: Part 2—Advanced topics," *Proc. IEEE*, vol. 85, no. 1, pp. 64–79, Jan. 1997.

[9] 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. Inf. Theory*, vol. 47, no. 1, pp. 446–450, Jan. 2001.

[10] 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. Inf. Theory*, vol. 48, no. 7, pp. 2105–2111, Jul. 2002.

[11] 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.

[12] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. Hoboken, NJ, USA: Wiley, 2003.

[13] R. R. Wilcox, *Introduction to Robust Estimation and Hypothesis Testing*. San Diego, CA, USA: Academic Press, 2005.
[14] D. N. Jayasimha, "Fault tolerance in multisensor networks," *IEEE Trans. Rel.*, vol. 45, no. 2, pp. 308–320, Jun. 1996.
[15] M. Staroswiecki, G. Hoblos, and A. Aïtouche, "Sensor network design for fault tolerant estimation," *Int. J. Adapt. Contr. Signal Process.*, vol. 18, pp. 55–72, 2004.
[16] Y. Ali and S. Narasimhan, "Redundant sensor network design for linear processes," *AIChE J.*, vol. 41, pp. 2237–2249, 1995.
[17] D. L. Donoho and P. J. Huber, "The notion of breakdown point," in *A Festschrift for Erich L. Lehman*, P. Bickel, K. Doksum, and J. L. Hodeges, Jr., Eds. Belmont, CA, USA: Wadsworth, 1983, pp. 157–184.
[18] J. J. Cho, Y. Chen, and Y. Ding, "Calculating the breakdown point condition of sparse linear models," *Technometrics*, vol. 51, no. 1, pp. 34–46, 2009.
[19] K. Kianfar, A. Pourhabib, and Y. Ding, "An integer programming approach for analyzing the measurement redundancy in structured linear systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 2, pp. 447–450, Apr. 2011.
[20] D. Lee and A. Lin, "Computational complexity of art gallery problems," *IEEE Trans. Inf. Theory*, vol. 32, no. 2, pp. 276–282, Feb. 1986.

Elif I. Gokce received her B.S. degree in industrial engineering from Istanbul Technical University, Istanbul, Turkey; her M.S. degree in industrial engineering from Sabancı University, Istanbul; and her Ph.D. degree in industrial engineering from Texas A&M University, College Station, TX, U.S.A. She is an operations research analyst at Bank of America, Fairfax, VA. Her research interests are in large-scale optimization, mixed integer programming, and cutting planes.

Abhishek K. Shrivastava received his B. Tech. (Hons.) degree in industrial engineering from Indian Institute of Technology, Kharagpur, India, in 2003; and his Ph.D. degree in industrial engineering from Texas A&M University in 2009. He is currently an Assistant Professor in the Department of Systems Engineering and Engineering Management at City University of Hong Kong. His research interests are in statistical modeling and analysis of complex systems, such as sensor networks, quality issues in nanomanufacturing, and data analytics for rare event diagnosis. He is a member of INFORMS, IIE, IMS, and HKSQ.

Yu Ding received his B.S. degree in precision engineering from the University of Science and Technology of China in 1993; his M.S. degree in precision instruments from Tsinghua University, China in 1996; his M.S. degree in mechanical engineering from Pennsylvania State University in 1998; and his Ph.D. degree in mechanical engineering from the University of Michigan in 2001. He is currently a Professor of Industrial and Systems Engineering and a Professor of Electrical and Computer Engineering at Texas A&M University. His research interests are in the area of systems informatics and control, and quality and reliability engineering. He currently serves as a department editor of IIE Transactions. He is a senior member of IEEE, and a member of INFORMS, IIE, and ASME.